GROMACS - Bug #2421

EwaldUnitTests and SimdUnitTests fail on ppc64le with gcc-8.0.1

02/25/2018 02:57 AM - Christoph Junghans

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>Mark Abraham</td>
</tr>
<tr>
<td>Category:</td>
<td>core library</td>
</tr>
<tr>
<td>Target version:</td>
<td>future</td>
</tr>
<tr>
<td>Affected version:</td>
<td>2018</td>
</tr>
<tr>
<td>Difficulty:</td>
<td>uncategorized</td>
</tr>
</tbody>
</table>

Description
   In item: /Forces/[0]/X
   Actual: -11.769416809082031
   Reference: -10.129964828491211
Difference: 1.63945 (1719090 single-prec. ULPs, rel. 0.162)
   Tolerance: abs. 1.43051e-06, 12 ULPs
Google Test trace:
and

Value of: pCopyDst[i]
   Actual: 5
Expected: pCopySrc[i]
   Which is: 6
SIMD load or store not moving data correctly for element 0

Found here: https://koji.fedoraproject.org/koji/taskinfo?taskID=25294220 (detailed build.log for ppe64le attached)

Associated revisions

Revision e6932745 - 03/19/2018 10:31 AM - Mark Abraham
Fix VSX SIMD with gcc 8
gcc 8 apparently generates different code for the former GROMACS code, which seems buggy. See
https://bugzilla.redhat.com/show_bug.cgi?id=1556989#c3
and https://gcc.gnu.org/bugzilla/show_bug.cgi?id=84907
Fixes #2421
Change-Id: I31492cd582b785cdfb42e8b999a165a7339ce4be

Revision f1712c37 - 03/27/2018 09:31 PM - Mark Abraham
More gcc-8 fixes for POWER
Fixes #2421
Change-Id: f43c13df3a217d0f2154dca0ef215ef9dcd27474a

Revision 9ae6059d - 06/13/2018 12:32 AM - Mark Abraham
More fixes to suit gcc 8 for double build
gcc 7 supports the same syntax, but only gcc 8 requires it.
xl code path untested, because we have no access to a working compiler

02/23/2020
Fixes #2421
Change-Id: i8f89af4b066be68e07a286a9fa45b8ded3c925f3

History
#1 - 02/25/2018 03:05 AM - Christoph Junghans
I think it is problem with gcc-8 as the same build worked fine on Fedora27 with gcc-7.2.1 (https://koji.fedoraproject.org/koji/buildinfo?buildID=1019225)

#2 - 02/25/2018 09:02 PM - Christoph Junghans
And this is with GMX_SIMD=IBM_VSX.

#3 - 02/25/2018 09:09 PM - Mark Abraham
gcc 8.0.1 is still pre-release. I don't have access to it anywhere, since my access is all to production login nodes. So I suggest we file a regression report with gcc.

#4 - 02/25/2018 09:41 PM - Christoph Junghans
You could do:

```bash
$ docker pull fedora:rawhide
$ docker run -it fedora:rawhide /bin/bash
```

#5 - 02/25/2018 10:53 PM - Mark Abraham
Christoph Junghans wrote:

```bash
You could do:
[...]
```

Yes, but that won't work on any machine capable of executing VSX instructions.

#6 - 02/26/2018 03:04 PM - Erik Lindahl
This appears to be bugs in the very basic load/store operations. Unfortunately I've seen those before, and although it might be possible to work around some of them, it indicates severe compiler bugs.

#7 - 03/16/2018 06:47 PM - Christoph Junghans
GCC upstream thinks this is a bug in Gromacs and proposed a patch here: https://bugzilla.redhat.com/show_bug.cgi?id=1556989#c3

```c
+++ gromacs-2018/src/gromacs/simd/impl_ibm_vsx/impl_ibm_vsx_simd_float.h 2018-03-16 17:02:45.915186989 +0100
@@ -121,14 +121,14 @@ static inline SimdFloat gmx_simdcall
     return {
         *reinterpret_cast<const __vector float*>(m)
         vec_xl(0, m)
-    };
+    vec_xl(0, m);
 }

static inline void gmx_simdcall
storeU(float *m, SimdFloat a)
{
-    *reinterpret_cast<__vector float*>(m) = a.simdInternal;
+    vec_xst(a.simdInternal, 0, m);
}

static inline SimdFloat gmx_simdcall
@@ -157,14 +157,14 @@ static inline SimdFInt32 gmx_simdcall
     return {
-        *reinterpret_cast<__vector int*>(m)
+        vec_xl(0, m)
         vec_xl(0, m);
-    };
+    vec_xl(0, m);
```

02/23/2020
static inline void gmx_simdcall
storeU(std::int32_t * m, SimdFInt32 a)
{  
    *reinterpret_cast<__vector int*>(m) = a.simdInternal_;  
+    vec_xst(a.simdInternal_, 0, m);  
}

static inline SimdFInt32 gmx_simdcall
 simdLoadU(const double *m, SimdDoubleTag = {})
{  
    return {  
-        *reinterpret_cast<const __vector double*>(m)  
+        vec_xl(0, m)  
    };
}

static inline void gmx_simdcall
storeU(double *m, SimdDouble a)
{  
-    *reinterpret_cast<__vector double*>(m) = a.simdInternal_;  
+    vec_xst(a.simdInternal_, 0, m);  
}

static inline SimdDouble gmx_simdcall
#8 - 03/16/2018 07:28 PM - Christoph Junghans
And http://gcc.gnu.org/PR84907

#9 - 03/16/2018 10:23 PM - Gerrit Code Review Bot
Gerrit received a related patchset '1' for Issue #2421.
Uploader: Mark Abraham (mark.j.abraham@gmail.com)
Change-Id: gromacs~release-2018~I31492cd582b7a165a7339ce4be
Gerrit URL: https://gerrit.gromacs.org/7688

#10 - 03/17/2018 12:38 AM - Mark Abraham
With and without that patch, gcc 7.2 in release mode is fine. gcc 6.3 without that patch is fine. But with it:

FAILED: src/gromacs/simd/tests/CMakeFiles/simd-test.dir/bootstrap_loadstore.cpp.o /gpfs/software/opt/gcc/6.3.0/bin/g++  
../src/gromacs/simd/tests/CMakeFiles/simd-test.dir/bootstrap_loadstore.cpp.o: In member function 'virtual void gmx::test::{anonymous}::SimdBootstrapTest_loadUI_Test::TestBody()':
../src/gromacs/simd/tests/bootstrap_loadstore.cpp:183:1: error: unrecognized insn:  
^  
insn 66 65 69 3 (set (reg:V4SI 270)  
(vec_select:V4SI (vec_select:V4SI (mem:V4SI (reg:DI 271 [ ivtmp.541 []]) [0 S16 A8])  
parallel |  
(const_int 3 [0x3])  
(const_int 2 [0x2])  
(const_int 1 [0x1])  
(const_int 0 [0]))  
})  
(parallel:V4SI [  
(const_int 2 [0x2])  
(const_int 3 [0x3])  
(const_int 0 [0])  
(const_int 1 [0x1]))  

02/23/2020
Based on the IBM feedback, it's only version 8 that will get the ABI change.

I would suggest that we simply add a check for the gcc version and use the old construct for version < 7.

Cheers,

Erik
Some of the other tests still fail after patching in https://github.com/gromacs/gromacs/commit/e693274554dfba67124d02769d0a885dae42b08

#17 - 03/20/2018 08:40 AM - Gerrit Code Review Bot
Gerrit received a related patchset ‘1’ for Issue #2421.
Uploader: Mark Abraham (mark.j.abraham@gmail.com)
Change-Id: gromacs~release-2018~I43c13df0a217d0f2154dca0e2f215ed9cd27474a
Gerrit URL: https://gerrit.gromacs.org/7696

#18 - 03/20/2018 09:15 AM - Mark Abraham
- Status changed from Resolved to Fix uploaded

#19 - 03/20/2018 02:19 PM - Mark Abraham
Double precision needs some work, too

#20 - 03/20/2018 04:23 PM - Mark Abraham
I think I have double fixed, but will finish it tomorrow

#21 - 03/21/2018 01:01 PM - Gerrit Code Review Bot
Gerrit received a related patchset ‘1’ for Issue #2421.
Uploader: Mark Abraham (mark.j.abraham@gmail.com)
Change-Id: gromacs~release-2018~I8f89af4b066be68e07a296a9fa5b8d3c925f3
Gerrit URL: https://gerrit.gromacs.org/7710

#22 - 03/21/2018 04:10 PM - Christoph Junghans
I patched 92d6e21 in, but it is still failing: https://koji.fedoraproject.org/koji/taskinfo?taskID=25859230

#23 - 03/21/2018 04:18 PM - Mark Abraham
- Target version changed from 2018.1 to 2018.2

#24 - 03/21/2018 04:20 PM - Mark Abraham
will look further, but am away for ~1 week

#25 - 03/27/2018 10:00 PM - Mark Abraham
- Status changed from Fix uploaded to Resolved

Applied in changeset f1712c3766eb80b52d1aab55b54d9b22cacfd68b.

#26 - 03/28/2018 09:43 AM - Christoph Junghans
Don't we need the 2nd patch set, too?

#27 - 03/28/2018 02:59 PM - Mark Abraham
- Status changed from Resolved to Fix uploaded

probably, that last post is done by a bot

#28 - 06/05/2018 04:44 PM - Mark Abraham
- Category changed from testing to core library
- Status changed from Fix uploaded to Feedback wanted

Christoph With gcc 8.1.0 on POWER9, I can no longer reproduce any issues with @make check (mixed and double; release and debug). Can you resubmit the fedora build and confirm? Thus I assume there is no further need of any fix.

#29 - 06/05/2018 04:46 PM - Mark Abraham
Mark Abraham wrote:

Christoph With gcc 8.1.0 on POWER9, I can no longer reproduce any issues with @make check (mixed and double; release and debug). Can you resubmit the fedora build and confirm? Thus I assume there is no further need of any fix.

Ignore this, I can reproduce the problem. I was mistakently checking the branch including the fix.

02/23/2020
#30 - 06/06/2018 02:06 AM - Christoph Junghans
- File build.log.txt added

Still persists with gcc-8.1.1, log attached

#31 - 06/07/2018 01:47 AM - Mark Abraham
Christoph Junghans wrote:

Still persists with gcc-8.1.1, log attached

OK, tests pass for me with gcc 8.1.0 on JURON with https://gerrit.gromacs.org/#/c/7710/3 - can you try that for us please Christoph?

#32 - 06/07/2018 02:41 PM - Christoph Junghans
- File build.log.txt added

Even patching ba13293 in didn't help.

#33 - 06/12/2018 05:38 PM - Christoph Junghans
Mark, I got a confused with all the patch flying around, I added https://gerrit.gromacs.org/#/c/7710/ (ba13293), do I need https://gerrit.gromacs.org/#/c/7688/ (e693274) as well?

#34 - 06/12/2018 05:47 PM - Mark Abraham
Christoph Junghans wrote:

Mark, I got a confused with all the patch flying around, I added https://gerrit.gromacs.org/#/c/7710/ (ba13293), do I need https://gerrit.gromacs.org/#/c/7688/ (e693274) as well?

No, the latter is the grandparent patch of the former - see git log

#35 - 06/12/2018 05:50 PM - Mark Abraham
I'll upload some comparable cmake+make logs shortly

#36 - 06/12/2018 10:44 PM - Mark Abraham
- File tmpi-double.txt added
- File marks-build-log.txt added
- Target version changed from 2018.2 to 2018.3

tmpi-double.txt is cut from Christoph's build.log.txt from comment 32. The only point of interest I can see in these build log diffs is that cmake host SIMD detection looks a bit different. So maybe both our observations are valid on the hardware+software stack we're testing. So I guess we have to defer resolving this issue until we have more information from somewhere.

#37 - 06/13/2018 12:45 AM - Mark Abraham
- Status changed from Feedback wanted to Resolved

Applied in changeset 9ae6059da61adba8fd4135446cbaa4ff21de682a.

#38 - 06/13/2018 04:23 PM - Paul Bauer
Hej, would you consider this one now as fully resolved, or should we still keep it open for the future?

#39 - 06/13/2018 10:57 PM - Christoph Junghans
I think the issue still persists on Fedora, but honestly I lost track, which patches got merged already.

#40 - 06/15/2018 11:55 AM - Paul Bauer
- Target version changed from 2018.3 to future

ok, I'll move this to future then, if there are no strong opinions against having it still open if more bugs are found

#41 - 01/03/2019 10:57 AM - Mark Abraham
- Status changed from Resolved to Closed

Open it when we have an issue. Otherwise we have hundreds of issues whose status is unknown!

<table>
<thead>
<tr>
<th>Files</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>build.log</td>
<td>35.2 MB</td>
<td>02/25/2018</td>
<td>Christoph Junghans</td>
</tr>
<tr>
<td>build.log</td>
<td>35.1 MB</td>
<td>03/20/2018</td>
<td>Christoph Junghans</td>
</tr>
<tr>
<td>build.log.txt</td>
<td>35.6 MB</td>
<td>06/06/2018</td>
<td>Christoph Junghans</td>
</tr>
<tr>
<td>build.log.txt</td>
<td>35.6 MB</td>
<td>06/07/2018</td>
<td>Christoph Junghans</td>
</tr>
<tr>
<td>marks-build-log.txt</td>
<td>1.48 MB</td>
<td>06/12/2018</td>
<td>Mark Abraham</td>
</tr>
<tr>
<td>tmpi-double.txt</td>
<td>6.83 MB</td>
<td>06/12/2018</td>
<td>Mark Abraham</td>
</tr>
</tbody>
</table>