BUG REPORT

January 19, 2017

On the bug of the Fortran compiler

The execution result malfunction was detected by executable created by Technical Computing Suite Fortran Compiler.

==

[PH12230]

1. Phenomena and Occurrence conditions
Unexpected execution result may be caused at the following conditions.

1) The -x option and one of the following compiler options are valid.
- -Ksimd (*1)
- -Kparallel
- -O3 (*2)

2) The source program includes subprograms (functions or subroutines).
3) The subprograms of 2) include loops.
4) The loops of 3) include more than one of the same internal subprogram calls or module subprogram calls.
5) The actual arguments of each internal subprograms or module subprograms of 4) are dummy arguments of the subprograms of 2).
6) About the same internal subprogram calls or module subprogram calls of 4), at least one of the actual arguments of the first call and the last call are the following expressions:
- An expression which is the access to the same variable (representing an array) with different subscripts.
- An expression which is the access to the different components of the same variable (representing a structure).

7) The internal subprograms or the module subprograms of 4) are inline expanded (*4).
8) The internal subprograms or the module subprograms of 4) include the references to the dummy arguments.
9) The internal subprogram calls or module subprogram calls which are different from 4)
exist after 8).
10) The internal subprograms or the module subprograms of 9) are not inline expanded (*4).

*1: Compiler option -O2 or higher induces -Ksimd.
*2: Compiler option -Kfast induces -O3.
*3: If there is no subscript, it is equal to the leading part of the arrays.
*4: Compiler message jwd8101o-i shows the inline expansion is applied.

<Example codes>
Option: -O2 -xtest.sub2

```
 program main
 use test
 real(8),dimension(8)  :: data1
 data data1/1,2,3,4,5,6,7,8/
 call sub1(data1,2)
 end program main

 module test
```

```
   contains
     subroutine sub1(data1,n)       ! 2)
     real(8),dimension(8)  :: data1
     integer :: i
     do i = 1, n                    ! 3)
       call sub2(data1(1))          ! 4),5),6),7)
       call sub2(data1(3))          ! 4),5),6),7)
     enddo
     end subroutine sub1

     subroutine sub2(data1)
     real(8),dimension(2)  :: data1
     data1(2) = data1(1)            ! 8)
     call sub3(data1(1))            ! 9),10)
     end subroutine sub2

     subroutine sub3(data3)
     real(8)   :: data3
     print *,data3
     end subroutine sub3
   end module test
```

2. Language
  Fortran

3. Cause
  The optimization to promote SIMD and automatic parallelization was incorrect, and an incorrect address may be set to the actual argument of the internal subprogram calls or module subprogram calls.

4. Possible preventive measures
  This bug can be avoided by specifying the compiler option -x0 (suppress the inline expansion).

5. Classification of detections
  This bug was detected in the field-SE

6. Scope of influence
  This is the bug of the Fortran compilers included in the following products.
    - Technical computing Suite V2.0L10 (FX100) DLIB and later

7. Correction delivery schedules
  The next patch will fix this problem.
  If the problem is detected by check tool, the emergency patch will be released immediately.

  Technical Computing Suite V2.0L20(FX100)
    Patch No.   : T01776-03
    Release plan : under coffering

8. Check tool
  The check tool delivery is scheduled at the mid of February in 2017.

===

Next Generation Technical Computing Unit
Language Development Div.
Development Dept.II