Debugging on Intel® Many Integrated Core Architecture – Lab Instructions

Lab 0 - Prerequisites

The lab assumes the work directory under /tmp/host-dbg/ - throughout this workshop we will have it in ~/workspace/debugger/. The Makefile compiles the executable prog and the shared library lib64/libmylib.so from main.cpp and mylib.cpp.

```
$ source /opt/intel/composer_xe_2013/bin/compilervars.sh intel64

$ make all

icpc -g -O0 -mmic -fPIC -c -c main.cpp

icpc -g -O0 -mmic -fPIC -c mylib.cpp

icc -mmic -shared mylib.o -o lib64/libmylib.so

icc -mmic main.o -L./lib64 -lmylib -o prog

x86_64-k1om-linux-ld: warning: libimf.so, needed by ./lib64/libmylib.so, not

found (try using -rpath or -rpath-link)

x86_64-k1om-linux-ld: warning: libsvml.so, needed by ./lib64/libmylib.so, not

found (try using -rpath or -rpath-link)

x86_64-k1om-linux-ld: warning: libintlc.so.5, needed by ./lib64/libmylib.so,
```

```
not found (try using -rpath or -rpath-link)
```

Lab 1 – MIC Native Debugging using IDB

Copy files to the coprocessor

Upload the application and shared library:

```
$ ssh mic0 mkdir –p /tmp/target-dbg/lib64
$ scp /tmp/host-dbg/prog mic0:/tmp/target-dbg/
$ scp /tmp/host-dbg/lib64/libmylib.so mic0:/tmp/target-dbg/lib64/libmylib.so
```

Upload the compiler shared libraries required for this example:

```
$ scp /opt/intel/composer_xe_2013/lib/mic/libintlc.so.5 mic0:/tmp/target-dbg/lib64/
$ scp /opt/intel/composer_xe_2013/lib/mic/libimf.so mic0:/tmp/target-dbg/lib64/
$ scp /opt/intel/composer_xe_2013/lib/mic/libsvml.so mic0:/tmp/target-dbg/lib64/
$ scp /opt/intel/composer_xe_2013/lib/mic/libirng.so mic0:/tmp/target-dbg/lib64/
```

Start the debugger, launch the debuggee

Start the debugger:

```
$ idbc_mic -tco -rconnect=tcpip:mic0:2000

Intel(R) Debugger Remote Server, Build [77.682.19]
This version is configured for k1om on Linux
Copyright (C) 2006-2011 Intel Corporation. All rights reserved.
Intel(R) Debugger for applications including Intel(R) MIC Architecture, Build [77.682.19]
NOTE: The evaluation period for this product ends in 585 days.
```

Set the remote file:

```
(idb) idb file-remote /tmp/target-dbg/prog
```

Setup the environment:

```
(idb) set env LD_LIBRARY_PATH /tmp/target-dbg/lib64
```

Set the symbol file:

```
(idb) file /tmp/host-dbg/prog
Reading symbols from /tmp/host-dbg/prog...done.
Info: Optimized variables show as <no value> when no location is allocated.
```

Set a breakpint at main, start the application:

```
(idb) break main
Breakpoint 1 at 0x400703: file /tmp/host-dbg/main.cpp, line 36.
(idb) run
Starting program: /tmp/host-dbg/prog
[New Thread 4614 (LWP 4614)]
Breakpoint 1, main (argc=1, argv=0x7fff9098b3a8) at /tmp/host-dbg/main.cpp:36
5       printPid();
```

Debug your application and…

```
(idb) quit
```

Lab 2 – MIC Native Debugging using IDB

Open another terminal and launch the debuggee on MIC:

```
$ ssh mic0
~ # cd tmp/target-dbg/
/tmp/target-dbg # export LD_LIBRARY_PATH=/tmp/target-dbg/lib64
/tmp/target-dbg # ./prog
PID: 3966
```

The debuggee prints its PID on the console. In another terminal we can instruct IDB to debug to that process:

```
$ idbc_mic -tco -rconnect=tcpip:mic0:2000

Intel(R) Debugger for applications including Intel(R) MIC Architecture, Build
[77.682.19]
NOTE: The evaluation period for this product ends in 585 days.

(idb) attach 3966 /tmp/host-dbg/prog

Attaching to program: /tmp/host-dbg/prog, process 3966
[New Thread 3966 (LWP 3966)]
Reading symbols from /tmp/host-dbg/prog...done.
__mktime_internal () in /lib64/libc-2.12.so
```

Set a breakpoint at the worker loop:

```
(idb) break main.cpp:38
Info: Optimized variables show as <no value> when no location is allocated.
Breakpoint 1 at 0x40070c: file /nfs/iul/disks/iul_team2/msturm/TPT/ISV-Workshop-lab/host-dbg/main.cpp,
line 38.

(idb) continue
Continuing.

Breakpoint 1, main (argc=1, argv=0x7fff92b35798) at /nfs/iul/disks/iul_team2/msturm/TPT/ISV-Workshop-
lab/host-dbg/main.cpp:38
38        while(loop)sleep(1);
```

Set a breakpoint outside the worker loop, set the loop variable to 0 and continue:

```
(idb) b main.cpp:39
Breakpoint 2 at 0x400729: file /nfs/iul/disks/iul_team2/msturm/TPT/ISV-Workshop-lab/host-dbg/main.cpp,
line 39.
(idb) p loop=0
$1 = false
(idb) c
Continuing.

Breakpoint 2, main (argc=1, argv=0x7fff92b35798) at /nfs/iul/disks/iul_team2/msturm/TPT/ISV-Workshop-
lab/host-dbg/main.cpp:39
39        printMsg("Test message");
(idb) c
Continuing.
Program exited normally.
```

“MSG: Test message” is printed to the other console and the application exits.