Release Notes: GNUARM-NONE-EABI v14.02

12th September 2014

KPIT Technologies Limited is releasing the GNUARM-NONE-EABI v14.02, a cross compiler tool for ARM based Renesas RZ micro-controllers.

The GNUARM-RZ toolchain has been superseded by GNUARM-NONE-EABI toolchain. The functionality is same as that of GNUARM-RZ, this is done to maintain the universal naming convention for ARM toolchains.

SALIENT FEATURES

1. The GNUARM-NONE-EABI v14.02 toolchain is based on following sources
   GCC 4.9.1 [released],
   Binutils 2.24 [released],
   Newlib 2.1.0 [released] and
   GDB 7.8 [released].

2. This is a generic ARM toolchain which has been tested for Cortex-A9, Cortex-R4 and Cortex-M3 targets.
   (Note: Toolchain has been tested only for `'little'` endian mode.)

3. The GNUARM-NONE-EABI toolchain comes with bug fixes and more code size optimization.

ABOUT GNUARM-NONE-EABI v14.02

Release Version:	GNUARM-NONE-EABI v14.02
Release Date:	12th September 2014
Platforms Supported:	Red Hat GNU/Linux v8.0 or later (or compatible distribution) Windows XP, Windows 7 (32-bit and 64-bit), Windows 8
Language:	C, C99, C++
Targets:	Cortex-A9, Cortex-R4 and Cortex-M3
Object File Format:	ELF

KPIT

CHANGES IN THIS RELEASE

This section describes the enhancements made and the issues fixed in the v14.02 release.

GCC:

1. The GNUARM-NONE-EABI toolchain generated incorrect bit access instruction with optimization option, '-O1' or higher. LDR instruction was generated instead of LDRH.

    This issue has been fixed.

INSTALLER and RPM:

1. The GNUARM-NONE-EABI Installer supports the 'Custom Installation' and 'Default Installation' modes. The "Default Installation" mode is set by default and will install the toolchain to a default location "C:\Program Files\KPIT". The username and activation key are silently accepted if cached in the registry. The "Custom Installation" mode allows user to install toolchain at custom directory.

2. The GNUARM-NONE-EABI toolchain supports integration with e2 studio (v3.1 version on-wards) IDE. Renesas e2 studio IDE will automatically detect the GNUARM-NONE-EABI toolchain on start-up for integration. Alternatively, you may also use the Toolchain Management feature in e2 studio IDE to achieve this.
    For details on e2 studio please refer the following link,
    http://www.renesas.com/products/tools/ide/ide_e2studio/index.jsp

    The GNUARM-NONE-EABI toolchain supports the '-mcpu' and the '-march' options.
    The '-march' option takes precedence for code generation and the '-mcpu' options is used for the CPU tuning. Please refer the test report for details on supported options for ARM, available at, www.kpitgnutools.com > Documentation > Test Reports > KPIT_GNUELF_v14.02_Tools_Test_Report > Please refer sections:
    'Compiler Options Testing' and
    'Assembler Options Testing'
    Warnings or errors may get generated in case there is a mismatch in flags on using these options.

3. The GNUARM-NONE-EABI v14.01 toolchain generates debugging information by default in the 'DWARF4' format, due to this hardware debugging fails by generating the following error:
    ```
    ERROR(CMD685-IMG54):
    ! Failed to load symbols for "RZ_R7S72100_LED_SAMPLE.x"
    ! Failed to demand load DWARF debugging information: section .debug_info, offset 0xb
    ```

    In order to debug a project successfully, user needs to explicitly pass '–gdwarf-2' option to generate the debugging information which is also understood by GDB debugger.

    For DS-5 IDE:
    User can add '–gdwarf-2' option to a project at the below location,
    Project > Properties > C/C++ Build > Settings > Tool Settings > GCC C Compiler > Miscellaneous > Other flags

    Note:

    There is no support to integrate the GNUARM-NONE-EABI v14.02 toolchain with DS-5 IDE. DS-5 IDE users would need to continue using the GNUARM-RZ v14.01 toolchain.

Users who would like to migrate the KPIT GNUARM-RZ projects created in DS-5 in to e2 studio v3.1.0.X IDE, can safely import the DS-5 GNUARM-RZ projects into the e2 studio IDE via the below feature, 'File' > 'Import' > 'DS-5 KPIT GNUARM-RZ Project'.
The imported DS-5 project would be automatically configured to use the latest integrated GNUARM-NONE-EABI toolchain with e2 studio IDE. Users can verify the option mapping between DS-5 and e2 studio.

Additionally, e2 studio projects created using the GNUARM-RZ toolchain would be automatically configured internally to use the GNUARM-NONE-EABI toolchain in e2 studio v3.1.0.X IDE.

KNOWN LIMITATIONS IN GNUARM-NONE-EABI

This section describes the known limitations in this release. We intend to fix these issues in our future releases.

We occasionally release maintenance packs for critical bug fixes.

Windows and GNU/Linux:

1. Assembler does not support ARM architectures `'armv5e'` and `'ep9312'`.

2. For following CPUs, compiler generates an instruction 'bx lr'.
   ```
   arm2, arm250, arm3, arm6, arm60, arm600, arm610, arm620, arm7, arm70, arm700,
   arm700i,        arm710, arm7100, arm710c,arm720, arm7500, arm7500fe, arm7d, arm7di,
   arm7dm, arm7dmi, arm7m.
   ```
   Assembler does not support this instruction for these CPUs, resulting in application build failure with assembler errors.
   Similar problem has been observed with the following architectures as well:
   ```
   armv2, armv2a, armv3, armv3m.
   ```

NEWLIB:

1. The 'C' library functions, `'strcat'` and `'strncat'` do not generate correct results with GDB simulator.

GDB:

1. For `'big endian'` targets debugger fails to start the debugging process.

2. For `'thumb'` mode:

   a. Debugger shows incorrect values for local and global variables.

   b. While stepping over the multiplication operation, debugging process goes in an infinite loop.

   c. While debugging floating point operations, execution halts with an error message:

   ```
   Unhandled v6 thumb insn: 4603
   0x00000004 in ?? ()
   ```

3. On using float-abi `'hard'`:

a. Debugging process halts with an error message:
```
0x00000004 in ?? ()
Cannot find bounds of current function
```

GDB-SIMULATOR:

1. The current ARM simulator port does not fully support the ARM-7 architecture.

   The run simulator fails to simulate the `'mul'` instruction correctly when destination register is identical to the first operand.

LIBGEN:

1. The GNUARM-NONE-EABI is a generic toolchain, however is tested only for Cortex-A9 core. The pre-built libraries generated using this toolchain, are also verified only for the Cortex-A9 core. However, user can dynamically create `'libm'` and `'libc'` libraries using the `'libgen'` tool by specifying correct compiler CPU options to generate libraries specific cores.

   While building the Newlib library using `'libgen'` tool, many compiler warnings are generated by the `'libgen'` tool. These warnings are displayed on `'stdout'` while building the runtime libraries. These warnings are observed while building the pre-built libraries as well.

2. The toolchain fails to link certain applications when project-build newlib libraries are used and generates the following errors:
```
./libout.a(syscalls.o): In function `_read':
syscalls.c:(.text+0x3c4): multiple definition of `_read'
./libout.a(linux-syscalls0.o):(.text+0x118): first defined here
./libout.a(syscalls.o): In function `_lseek':
syscalls.c:(.text+0x638): multiple definition of `_lseek'
```

   Workaround:
   User would need to use pre-built newlib libraries in this case.

TOOLS INFORMATION

1. The optimized libraries provided along with the newlib libraries in the toolchain do not require a separate download.

2. The optimized libraries (`'liboptm.a'` and `'liboptc.a'`) are not provided under GNU GPL. The source code of these optimized libraries is neither released nor available on request.

3. The `"libgen"` utility is not provided under GNU GPL. The source code of the `"libgen"` utility is neither released nor available on request.

For free technical support, please register at: http://www.kpitgnutools.com
For your feedback and suggestions, please visit: http://www.kpitgnutools.com/feedback.php