Release Notes: GNUARM-RZ v14.01

28th February 2014

KPIT Technologies Limited is pleased to release the GNUARM-RZ v14.01, a cross compiler toolchain for ARM based Renesas RZ micro-controllers.

SALIENT FEATURES

1. The GNUARM-RZ v14.01 toolchain is based on following sources
    GCC 4.8.2 [released],
    Binutils 2.24 [released],
    Newlib 2.1.0 [released] and
    GDB 7.6.2 [released].

2. This is a generic ARM toolchain which has been tested for Cortex-A9 target.
    (NOTE: Toolchain has been tested only for `little` endian mode.)

3. The `'return'` code statement did not work correctly due to internal error in GDB. This is now fixed.

4. On certain Windows 7 machines, library generator used to crash. This is now fixed.

5. The GNUARM-RZ v14.01 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.

6. The GNUARM-RZ toolchain supports integration with e2 studio (v2.1 version on-wards) and DS-5 (v5.15) IDEs.
    Renesas e2 studio IDE will automatically detect the GNUARM-RZ v14.01 toolchain on start-up for registration. 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

ABOUT GNUARM-RZ v14.01

Release Version:	GNUARM-RZ v14.01
Release Date:	28th February 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

Object File Format:	ELF

CHANGES IN THE GNUARM-RZ V14.01 RELEASE

This section describes the enhancements made and the issues fixed in the v14.01 release.

GCC:

1. The GNUARM-RZ 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 below location
Project-> Properties-> C/C++ Build-> Settings-> Tool Settings-> GCC C Compiler-> Miscellaneous-> Other flags

KNOWN LIMITATIONS IN ARM-RZ-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.
    ```

KNOWN PROBLEMS IN ARM-RZ-EABI

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

GDB:
1. For 'big endian' targets debugger fail 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 run simulator fails to simulate the 'mul' instruction correctly when destination register is identical to the first operand.

LIBGEN:

1. 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.

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