

Release Notes about Adding new toolchain to Itib on i.MX platform

Table of Contents

1	Release History.....	3
2	Introduction.....	4
2.1	Generate rpm package.....	4
2.2	Update the toolchain.lkc	5
2.3	Verification the build process	8
2.3.1	Installation in Ltib.....	8
2.3.2	Build options.....	8
2.4	Known Issues	9

1 Release History

RELEASE NUMBER	DELIVERABLES	FEATURES
1.0	<ul style="list-style-type: none">• Documentation• Porting process• Build options in different SoCs	<ul style="list-style-type: none">• Rpm generation of the toolchain• Add new toolchain to the ltib
2.0		<ul style="list-style-type: none">• Add known issues• How to change default tc from gcc4.3.3 to gcc4.1.2

2 Introduction

This document aims at how to add a new toolchain to the Ltib. Totally, the process consists of three parts:

- Generate the rpm package of the new toolchain
- Edit the toolchain.lkc and add the new toolchain item selection
- Verify the build result and adjust the gcc compile flags

Next, we will use code sourcery gcc toolchain as one example.

2.1 Generate rpm package

At first, get the compressed package of the GNU/Linux toochain from the below web address (arm-2009q1-203-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.tar): <http://www.codesourcery.com/sqpp/lite/arm/portal/release858>

In order to follow standard path convention for toolchains, we should reconstruct/rename the package like this:

```
tar cvf arm-2009q1-203-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.tar
mkdir gcc-4.3.3-glibc-2.8-cs2009q1-203
cd gcc-4.3.3-glibc-2.8-cs2009q1-203
mkdir arm-none-linux-gnueabi
cd arm-none-linux-gnueabi
cp -rf ../../arm-2009q1/* .
```

Since the package is much larger than FSL toolchains, we remove the three locale folders to decrease the code sourcery toolchain size. (libc/*/usr/lib/locale) FSL built toolchains don't include the locale folders. Now you may repack the modified toolchain:

```
tar cvzf gcc-4.3.3-glibc-2.8-cs2009q1-203.tar.gz gcc-4.3.3-glibc-2.8-cs2009q1-203
```

Next we will create one *.spec file to generate the rpm package. Here we name the file as toolchain.spec. The spec file would include:

```
Summary: x86-linux hosted GCC arm-none-linux-gnueabi cross compiler
Name: gcc-4.3.3-glibc-2.8-cs2009q1
Version: 203
Release: 1
Source0: gcc-4.3.3-glibc-2.8-cs2009q1-203.tar.gz
License: GPL
Group: Programming/Compiler
%description
x86-linux hosted GCC arm-none-linux-gnueabi cross compiler
%prep
%setup -q
#%build
#./configure
```

```
#make
%install
mkdir /opt/freescale/usr/local/gcc-4.3.3-glibc-2.8-cs2009q1-203
cp -rf * /opt/freescale/usr/local/gcc-4.3.3-glibc-2.8-cs2009q1-203
%files
%defattr(-,root,root)
/opt/freescale/usr/local/gcc-4.3.3-glibc-2.8-cs2009q1-203
On ubuntu server, copy the gcc-4.3.3-glibc-2.8-cs2009q1-203.tar.gz to /usr/src/
rpm/SOURCES directory (the rpm-build command will find the target package
from the folder by default). Now you can run rpmbuild command:
cp gcc-4.3.3-glibc-2.8-cs2009q1-203.tar.gz /usr/src/ rpm/SOURCES
rpmbuild -bb toolchain.spec
```

This may take several minutes, after the rpm package is created, you can find it in /usr/src/rpm/RPMS/i386/gcc-4.3.3-glibc-2.8-cs2009q1-203-1.i386.rpm.

Also you can use rpm command to query if the rpm package is created correctly:

```
rpm -qpl /usr/src/rpm/RPMS/i386/gcc-4.3.3-glibc-2.8-cs2009q1-203-1.i386.rpm
```

The content should be the same with the uncompressed gcc-4.3.3-glibc-2.8-cs2009q1-203.tar.gz.

2.2 Update the toolchain.lkc

After you prepare the rpm toolchain package, you can try the build in Ltib now. Remember to copy the **gcc-4.3.3-glibc-2.8-cs2009q1-203-1.i386.rpm** to /opt/Freescale/pkgs directory. When you add a new toochain into Ltib, you should edit the config/userspace/toolchain.lkc file. Here is the example showing how to add the new toolchain and it is described by one patch:

```
diff --git a/config/userspace/toolchain.lkc b/config/userspace/toolchain.lkc
index a1a46e2..17ae2be 100644
--- a/config/userspace/toolchain.lkc
+++ b/config/userspace/toolchain.lkc
@@ -541,6 +541,16 @@ if ARMEABI_ARCH
    list of choices select the custom option and enter the
    toolchain path and prefix in the appropriate boxes.

+ config TOOLCHAIN_ARMEABI_CODE_SOURCERY
+ bool "ARMv5T,ARMv6, gcc-4.3.3,Multi-lib,gnueabi/glibc-2.8" if GLIBC || LIBC_NONE ||
LIBC_HACKING
+ help
+ gcc-4.3.3 glibc-2.8 multilib armeabi toolchain.
+ Optimized for armv5t,armv6 cores.
```

```

+     Library versions available for:
+     - Soft float      (Default)
+     - Hard float, VFP, NEON (-mfloat-abi=softfp -mfp=neon -mfp=vfp)
+     Toolchain built by Freescale.
+
+     config TOOLCHAIN_ARMEABI_GLIBC1
+         bool "ARMv5te gcc-4.1.2,Multi-lib,gnueabi/glibc-2.5-nptl-3" if GLIBC || LIBC_NONE ||
LIBC_HACKING
+         help
@@ -656,6 +666,7 @@ config TOOLCHAIN
+         default freescale-coldfire-m68k-uclinux-4.2-153.i686.rpm if
TOOLCHAIN_68KNOMMU_UCLIBC2
+         default freescale-coldfire-m68k-uclinux-4.3-45.i686.rpm if
TOOLCHAIN_68KNOMMU_UCLIBC3
+         default freescale-coldfire-m68k-linux-gnu-4.3-43.i686.rpm if TOOLCHAIN_68K_GLIBC1
+         default gcc-4.3.3-glibc-2.8-cs2009q1-203-1.i386.rpm if
TOOLCHAIN_ARMEABI_CODE_SOURCERY
config TOOLCHAIN_PATH
+         string "Enter the custom toolchain path." if CUSTOM_TOOLCHAIN
@@ -685,6 +696,7 @@ config TOOLCHAIN_PATH
+         default "/opt/freescale/usr/local/gcc-4.2.153-uclibc-0.9.153/m68k-uclinux" if
TOOLCHAIN_68KNOMMU_UCLIBC2
+         default "/opt/freescale/usr/local/gcc-4.3.45-uclibc-0.9.45/m68k-uclinux" if
TOOLCHAIN_68KNOMMU_UCLIBC3
+         default "/opt/freescale/usr/local/gcc-4.3.43-eglibc-2.5.43/m68k-linux" if
TOOLCHAIN_68K_GLIBC1
+         default "/opt/freescale/usr/local/gcc-4.3.3-glibc-2.8-cs2009q1-203/arm-none-linux-gnueabi" if
TOOLCHAIN_ARMEABI_CODE_SOURCERY
config CUSTOM_TOOLCHAIN_PREFIX
+         depends CUSTOM_TOOLCHAIN
@@ -705,7 +717,7 @@ config TOOLCHAIN_PREFIX
+         default "powerpc64-linux-" if TOOLCHAIN_PPC64_GLIBC1
+         default "powerpc-823-linux-" if TOOLCHAIN_8XX_GLIBC2
+         default "arm-linux-" if TOOLCHAIN_ARMOABI_GLIBC1
-         default "arm-none-linux-gnueabi-" if TOOLCHAIN_ARMEABI_GLIBC1
+         default "arm-none-linux-gnueabi-" if TOOLCHAIN_ARMEABI_GLIBC1 ||
TOOLCHAIN_ARMEABI_CODE_SOURCERY
+         default "m68k-uclinux-" if TOOLCHAIN_68KNOMMU_UCLIBC1 ||
TOOLCHAIN_68KNOMMU_UCLIBC2 || TOOLCHAIN_68KNOMMU_UCLIBC3
@@ -724,8 +736,8 @@ config TOOLCHAIN_PREFIX

```

```

default "-msoft-float -mcpu=860" if TOOLCHAIN_8XX_GLIBC1 || TOOLCHAIN_8XX_GLIBC2
|| TOOLCHAIN_8XX_UCLIBC1 || TOOLCHAIN_8XX_GLIBC3

default "" if TOOLCHAIN_E300_GLIBC1 || TOOLCHAIN_E300_UCLIBC1 ||
TOOLCHAIN_PPC64_UCLIBC1 || TOOLCHAIN_E300_GLIBC2

default "-O2 -fsigned-char" if TOOLCHAIN_ARMOABI_GLIBC1

- default "-O2 -fsigned-char -mfloat-abi=softfp -mfpv=vfp" if ( VFP_ARCH &&
TOOLCHAIN_ARMEABI_GLIBC1 )

- default "-O2 -fsigned-char -msoft-float" if ( ! VFP_ARCH &&
TOOLCHAIN_ARMEABI_GLIBC1 )

+ default "-O2 -fsigned-char -mfloat-abi=softfp -mfpv=vfp" if ( VFP_ARCH &&
(TOOLCHAIN_ARMEABI_GLIBC1 || TOOLCHAIN_ARMEABI_CODE_SOURCERY))

+ default "-O2 -fsigned-char -msoft-float" if ( ! VFP_ARCH &&
(TOOLCHAIN_ARMEABI_GLIBC1 || TOOLCHAIN_ARMEABI_CODE_SOURCERY))

default "-mcpu=5208 -DCONFIG_COLDFIRE" if TOOLCHAIN_68KNOMMU_UCLIBC2 &&
( PLATFORM = m520xevb )

```

If you want to enable the newtoolchain as the default build toolchain. Taking the i.MX51 as the example, the config/platform/imx/imx51.cf should be modified as bellows:

```

--- a/config/platform/imx/imx51.cf
+++ b/config/platform/imx/imx51.cf
@@ -55,12 +55,13 @@ CONFIG_SYS_WANT_SHARED=y
#
# Toolchain selection.
#
-CONFIG_TOOLCHAIN_ARMEABI_GLIBC1=y
+CONFIG_TOOLCHAIN_ARMEABI_CODE_SOURCERY=y
+# CONFIG_TOOLCHAIN_ARMEABI_GLIBC1 is not set
# CONFIG_TOOLCHAIN_ARMEABI_CUSTOM is not set
-CONFIG_TOOLCHAIN="tc-fsl-x86lnx-armeabi-nptl-4.1.2-3.i386.rpm"
-CONFIG_TOOLCHAIN_PATH="/opt/freescale/usr/local/gcc-4.1.2-glibc-2.5-nptl-3/arm-none-linux-
gnueabi"
+CONFIG_TOOLCHAIN="gcc-4.3.3-glibc-2.8-cs2009q1-203-1.i386.rpm"
+CONFIG_TOOLCHAIN_PATH="/opt/freescale/usr/local/gcc-4.3.3-glibc-2.8-cs2009q1-203/arm-none-
linux-gnueabi"
CONFIG_TOOLCHAIN_PREFIX="arm-none-linux-gnueabi-"
-CONFIG_TOOLCHAIN_CFLAGS="-O2 -fsigned-char -mfloat-abi=softfp -mfpv=vfp"
+CONFIG_TOOLCHAIN_CFLAGS="-O2 -fsigned-char -mfloat-abi=softfp -mfpv=neon -march=armv6"

```


2.4 Known Issues

When changing the default toolchain of i.MX platform to gcc 4.3.3, we found some performance issue on the code sourcery 4.3.3. Such as the memcpy function in libc is not optimized as FSL tc 4.1.2. After disassemble the memcpy function, we found it just use ldrb to perform the data transfer. So in the 09.12.00 release, we restore to the gcc 4.1.2 toolchain due to the performance issue.

- To roll back to the gcc4.1.2 FSL toolchain, one kernel patch is reverted for MX51 platform:

```
git revert 39b2ec2cfc591472db51c75944d8462255e9cf4d
```

```
#0001-i.MX-Restore-tc-selection-to-gcc4.1.2-due-to-perfor.patch
```

- The restore operation in ltib will be done in the each config file under config/platform subfolder. Following up the patch can roll back to FSL gcc 4.1.2 toolchain in Ltib.



WordPad Document