iMX6 Series Processor

EC / NSD / SBC 2100

Linux Toolchain User Guide

RELEASE NOTES

Version	Date	Notes
1.00	Oct-31-2013	Initial release
2.00	Jan-20-2014	Edited by technical writer

Table of Contents

1. Introduction

Toolchain is a set of software development tools that are used to create a piece of software. It consists of a text editor for editing source code, a compiler, and a linker to transform the source code into an executable program. It can also create libraries to provide interfaces to the Linux 3.0.35 platform.

You can use Toolchain to compile your existing source code and be a cross-platform development tool, to let users implement and operate their source code on multiple computing platforms. Toolchain from ICNEXUS includes some general libraries like JPGlib, PNGlib, Zlib, and FreeTpyelib for you to use during development.

2. Operating System and Development Environment

System Requirements:

- IBM-compatible PC
- 1.6 GHz CPU or higher
- 30 GB free disk space or more
- 512 MB RAM or more
- Linux-type OS

3. Setting up Toolchain

3.1 Installation

To install Toolchain on your Linux system, please do the following steps. First, extract toolchain\ armv6_codesourcery.tar.bz2.

```
$ cd /opt/
$ sudo tar –vxjf fsl-linaro-toolchain.tar.bz2
$ ls fsl-linaro-toolchain
   arm-fsl-linux-gnueabi   bin   include   lib   libexec   native   share
```

Next, open the file .bashrc and add '**export PATH=/opt/fsl-linaro-toolchain/bin$PATH:**' at the end of the file. This will register the Toolchain environment on the Linux system.

3.2 Environment Check

Check the version of compiler you are using.

```
$arm-none-linux-gnueabi-gcc –v
Using built-in specs.
Target: arm-none-linux-gnueabi
Configured with: ……
…..
Thread model: posix
gcc version 4.3.2 (Sourcery G++ Lite 2008q3-72)
```

4. Writing and Compiling in C

4.1 C Programming

Make a simple .c file.

```
#include <stdio.h>
int main(int argc,char* argv[])
        {
                printf("Hello ICNexus!!\n");
                return(0);
        }
```

4.2 Cross Compiling

Write a simple Makefile.

```
all:hello.c
            arm-none-linux-gnueabi-gcc hello.c -o hello
clean:
            rm -f hello
```

Compile the C file.

```
$ ls
hello.c    Makefile
$ make
arm-none-linux-gnueabi-gcc hello.c -o hello
$ ls
hello    hello.c    Makefile
```

Copy the hello executable file to the Micro SD card and insert it into NSD device.
Execute the file you have compiled as following:

```
# cd /mnt/SD
# ./hello
Hello ICNexus!!
```