Freescale Semiconductor
Application Note

Document Number: AN4808

Kinetis GCC Build Tools Options for Optimal Performance on the Cortex-M0+/M4 Core

1 Introduction

This document describes two sets of options that can be used with the CodeWarrior tools for optimal performance on the Cortex-M0+ and M4 cores. One set optimizes the execution speed; another set optimizes the size. You can use the build tools options described in this document for optimal performance, but the build tools settings must be set according to the application being developed.

For more information on the Kinetis GCC Compiler, refer to CodeWarrior Development Studio for Microcontrollers V10.x Kinetis GCC Build Tools Reference Manual by Freescale.

Contents

2 Optimization for Speed

Following are the options for speed optimization:

- Compiler Options
- Linker Options

2.1 Compiler Options

In order to reduce the overall cycle count of the code, pass the following options to the compiler:

```
-mcpu=cortex-m0plus/m4 -mlittle-endian -mthumb -g3 -nostdinc -include
libm.prefix -include lib_c++.prefix -std=c99 -O2 -finline-functions -
ffunction-sections -fdata-sections
```

2.2 Linker Options

In order to reduce the overall cycle count of the code, pass the following options to the linker:

```
-mcpu=cortex-m0plus/m4 -mlittle-endian -mthumb -nostartfiles -
nodefaultlibs -nostdlib  -Wl,--gc-sections -Wl,--undefined=sys_exit -
Wl,-n -Wl,-Map=test.map  -Wl,--start-group -lstdc++ -lrt -lsupc++ -lc99
-lm -lgcc -lhosted -Wl,--end-group
```

3 Optimization for Size

Specific options direct the compiler and linker to optimize the generated code for smaller size. The following sections provide details on these strategies.

3.1 Compiler Options

```
-mcpu=cortex-m0plus/m4 -mlittle-endian -mthumb -g3 -nostdinc -include
libm.prefix -include lib_c++.prefix -std=c99 -Os -ffunction-sections -
fdata-sections
```

3.2 Linker Options

```
-mcpu=cortex-m0plus/m4 -mlittle-endian -mthumb -nostartfiles -
nodefaultlibs -nostdlib  -Wl,--gc-sections -Wl,--undefined=sys_exit -
Wl,-n -Wl,-Map=test.map  -Wl,--start-group -lstdc++ -lrt -lsupc++ -lc99
-lm -lgcc -lhosted -Wl,--end-group
```

NOTE Use `--mcpu=cortex-m0plus` or `-mcpu=cortex-m4` depending on the core instead of `-mcpu=cortex-m0plus/m4`.

NOTE Use `-mfpu=fpv4-sp-d16  -mfloat-abi=hard` in the compiler options for optional cortex-m4 FPU applications

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

Document Number: AN4808

12 September 2013