NXP GCC 优化问题的总结

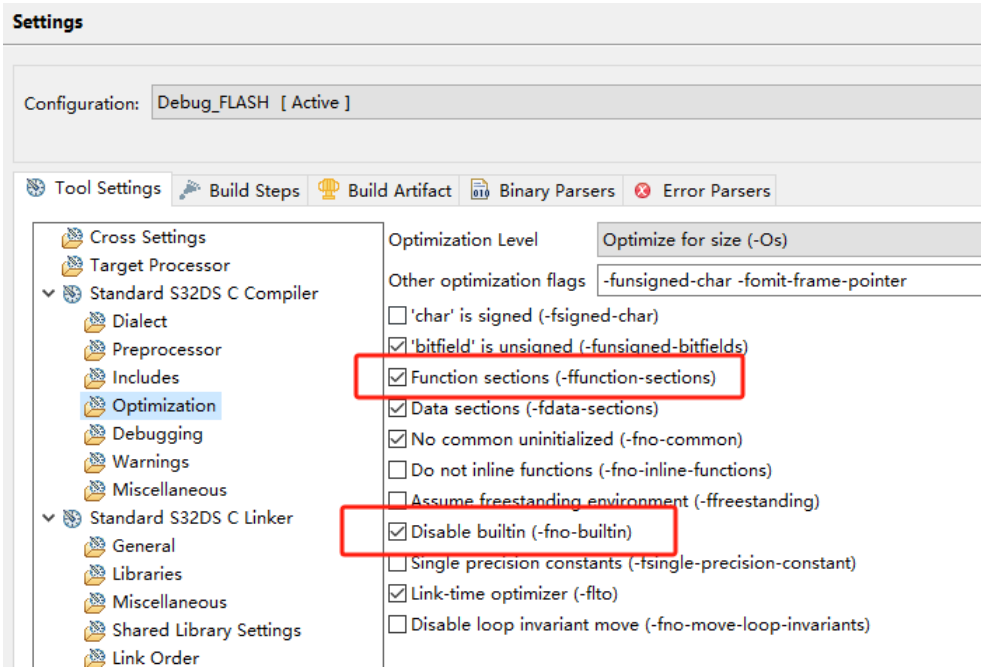
Peter Chen（Peter_Chen@wtmec.com）

1. 现状

就 2024 年年底而言，NXP 提供给客户用于 S32K3 开发软件是 S32DS for S32 Platform。IDE 自带一个免费的 GCC 编译器，用于生成代码。

版本	说明
6.3.1 build 1620	用于编译 S32K1 SDK 4.x 版本
9.2 build 1649	用于编译 S32K3 RTD 1.0.0 版本
10.2 build 1728	主力版本，用于编译 S32K3 RTD 2.0.0 – 5.0.0 版本
11.4 build 1763	新版本，尚未与特定版本 RTD 相关联

arm-none-eabi-gcc 的原始版本是不支持使用#pragma 指令来指定一个区块的默认段的设置。但是 MCAL 的代码大部分的函数和变量的定义需要使用#pragma GCC section 来设置段名。NXP 给出的方法是修改 GCC 的源代码，重新编译后发布，具体修改的内容可以参考 NXP GCC 安装目录中 patches_applied 中的 patch 文件。现在的修改代码有问题，导致现在 NXP GCC 的优化功能有问题。NXP GCC v10.2 中-ffunction-sections 功能无法使用，NXP GCC v11.4 中-ffunctino-sections 功能可以使用了，但是-flto 功能无法使用。



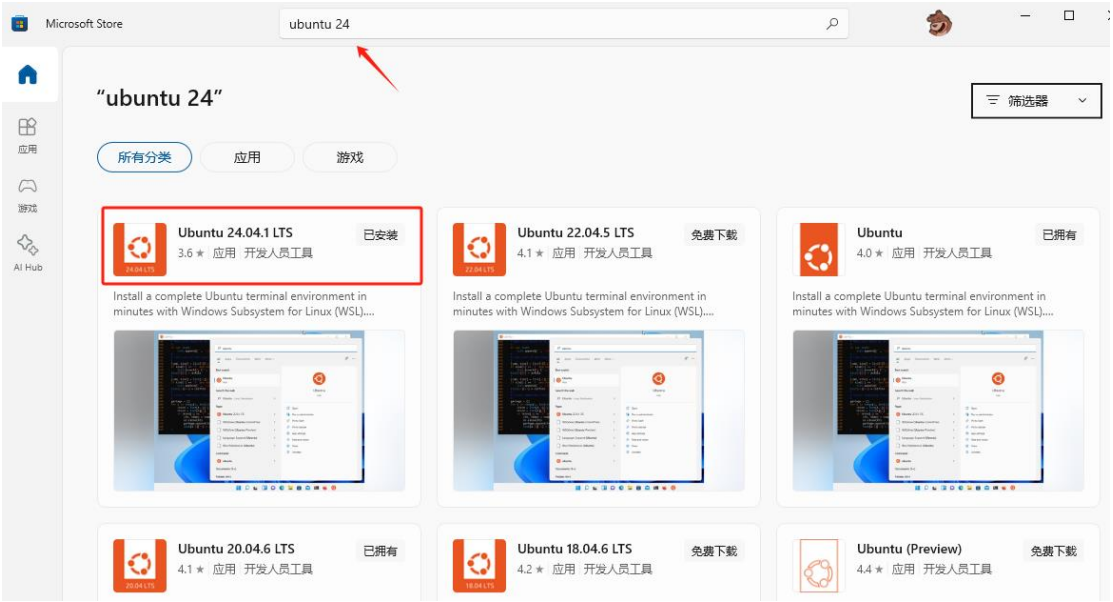
ARM 官方出的 Toolchain arm-none-eabi-gcc 已经发布到 v14.2 了，但是由于不支持

#pragma GCC section 语法，我们无法直接使用，暂时只能使用 NXP 提供的魔改版本。

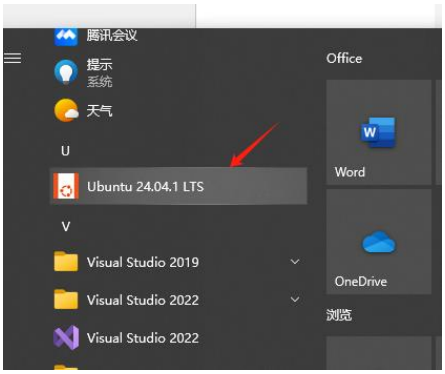
2. 解决方法

现在 GCC 是支持插件系统的，笔者根据 GCC 的规范制作了一个 GCC 的插件（ https://github.com/chenzch/GCC_Section_Plugin ），用于让标准版本的 arm-none-eabi-gcc 支持#pragma GCC section 的语法。测试以后优化选项-ffunction-sections 和 -flto 均可正常使用。但是因为没有找到在 Windows 下编译插件的方法，现在这个插件只能运行在 Linux 环境下。下文将介绍如何在 Windows 系统中的 WSL 里安装并使用这个插件，让现有的 S32K3 的程序达到最大的优化。

1. 在 Windows 上通过应用商店安装 Ubuntu 24.04.1。



安装完以后，通过开始菜单运行 Ubuntu，如果启动有问题，需要以管理员权限运行命令 wsl --update，用以更新 wsl 子系统。

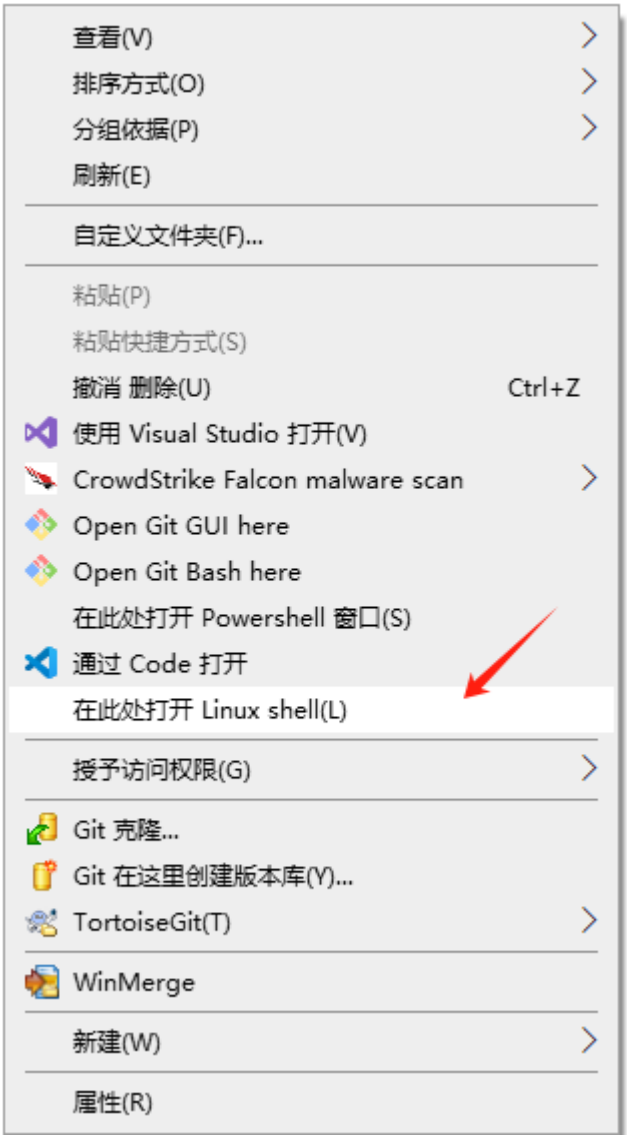


2. 启动 Linux 以后安装 arm-none-eabi-gcc。

在 ubuntu 系统的命令行中运行 sudo apt update 进行目录的更新。更新结束以后，运行 sudo apt install gcc-arm-none-eabi 来安装原始版本的 gcc，随 Ubuntu 24.04.1 发布的是 v13.2。

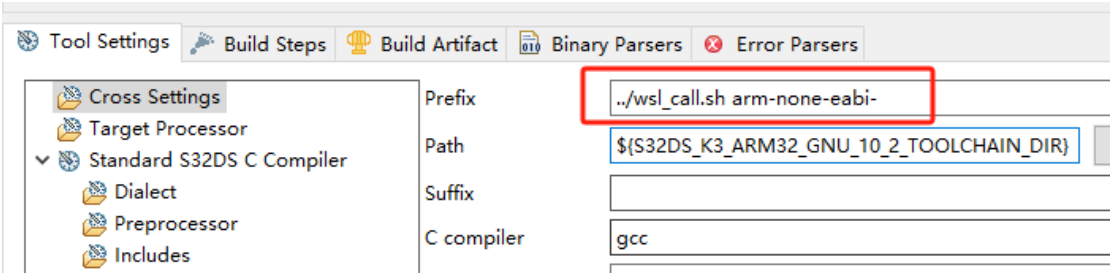
3. 将插件文件复制到 arm-none-eabi-gcc 的插件目录下

下载项目中的 gccsection13.so 和 wsl_call.sh 这 2 个文件，并在下载的目录中，按住 shift 按键并右单击目录空白处，选择“在此处打开 Linux shell”。然后在弹出的命令行中输入以下命令，cp gccsection13.so `arm-none-eabi-gcc -print-file-name=plugin`，注意这里最后一个参数的 arm-none……=plugin 两端的不是单引号，是 tab 按键上面的反单引号。gccsection13.so 是事先编译好的针对 gcc13 的插件。用户也可以使用项目里的源代码和 g++进行编译。cp 程序运行以后，插件就复制到 gcc 的插件目录中了。



4. 将 wsl_call.sh 文件复制到项目根目录中并将项目属性里 Cross Settings 的 Prefix 设

置前增加一个../wsl_call.sh，修改如下图。



保存以后，就可以使用 GCC 设置页面里的所有优化选项了。另外 wsl_call.sh 中 35 行会自动给编译器添加一个-fplugin 的选项，如果第 3 步里复制到 GCC 插件目录下的 so 文件改过名字了，则需要手动修改这一行后面跟随的插件名称，插件名称就是文件名去掉.so。