NXP GCC优化问题的总结

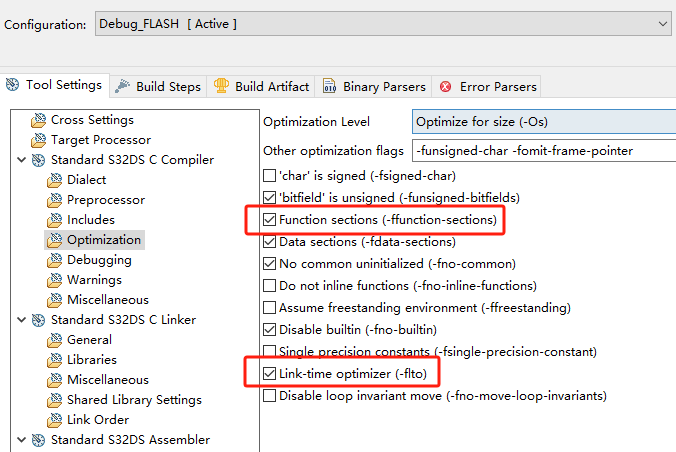
Peter Chen（[Peter\_Chen@wtmec.com](mailto:Peter_Chen@wtmec.com)）

# 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。

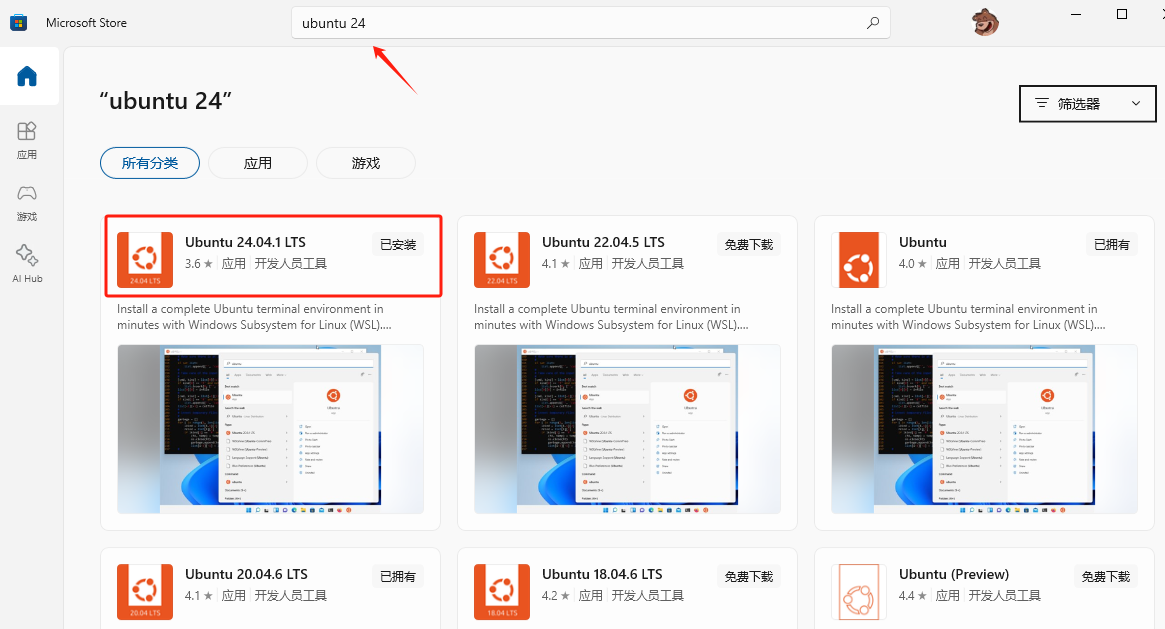


ARM官方出的Toolchain arm-none-eabi-gcc 已经发布到v14.2了，但是由于不支持#pragma GCC section语法，我们无法直接使用，暂时只能使用NXP提供的魔改版本。

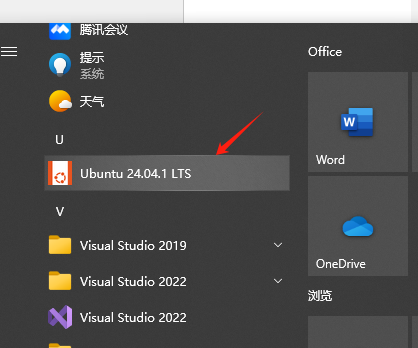
# 解决方法

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。

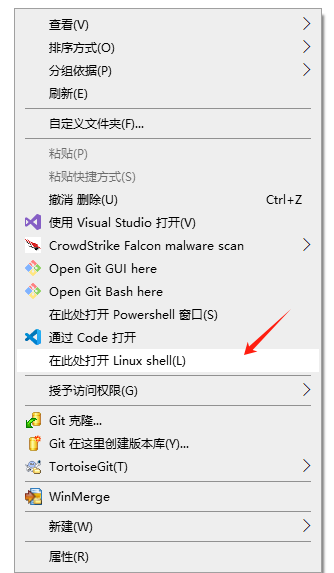


1. 启动Linux以后安装arm-none-eabi-gcc。

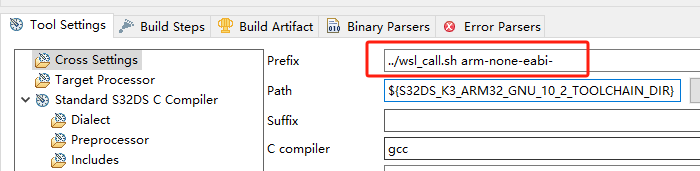
在ubuntu系统的命令行中运行sudo apt update进行目录的更新。更新结束以后，运行sudo apt install gcc-arm-none-eabi来安装原始版本的gcc，随Ubuntu 24.04.1发布的是v13.2。

1. 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。



1. 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。

