目录

[1 M2351中TrustZone概括 1](#_Toc498525784)

[2 工程创建 2](#_Toc498525785)

[2.1 创建工程 2](#_Toc498525786)

[2.2 工程说明 3](#_Toc498525787)

[2.3 TrustZone程序说明 5](#_Toc498525788)

[2.4 LiteOS移植 5](#_Toc498525789)

[2.5 程序烧录步骤， 6](#_Toc498525790)

# M2351中TrustZone概括

在armv8-m具有新功能称为TrustZone。TrustZone添加一个额外的安全状态让两安全水平完全隔离。Cortex M23处理器架构设计的额外的安全状态，与硬件级别的保护措施相结合，使得正常的软件（非安全状态）不可能在非安全状态下访问或修改信息（包括指令和数据）。为了定义处理器的当前状态，内存映射需要划分为安全内存和非安全内存空间。

1. 处理器内部的有一个硬件单元称为安全属性单元(SAU)。这个单元是可编程的，并且有一个类似于内存保护单元（MPU）的编程模型。安全软件可以通过编程SAU来定义内存空间的安全设置。SAU可由芯片设计者配置。
2. 属性单元把内存空间分配为以下三个部分：

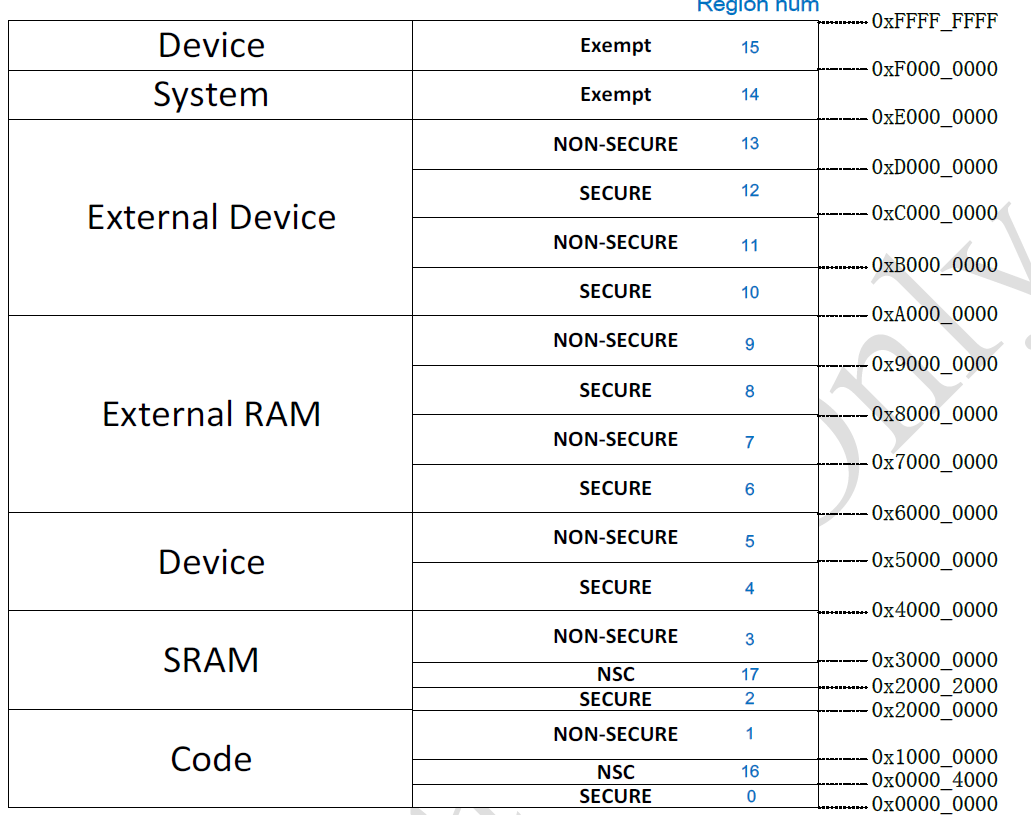
安全区：包括安全编程代码和数据（包括安全堆栈和其他任何安全数据）；

非安全呼叫区(NSC)：包括非安全区调用安全区函数的入口函数（例如APIs的入口指针）；

非安全区：包括安全编程代码和数据；

处理器状态依赖于内存空间定义：当处理器在安全区域运行代码时，它处于安全状态，否则它处于非安全状态。应用程序代码可以调用另一域中分支/代码，处理器自动检测安全域开关。

内存分配图



# 工程创建

## 创建工程

工程创建可以用新塘提供的M2351\_TZTG\_Tool\_Beta工具，该工具是可视化的，可以通过鼠标拖拽来设定安全区ROM和RAM区大小（图2-1）具体有以下功能：设置安全区 ROM 的大小；设置安全区 SRAM 的大小；产生安全区项目模板；产生飞安全区项目模板。

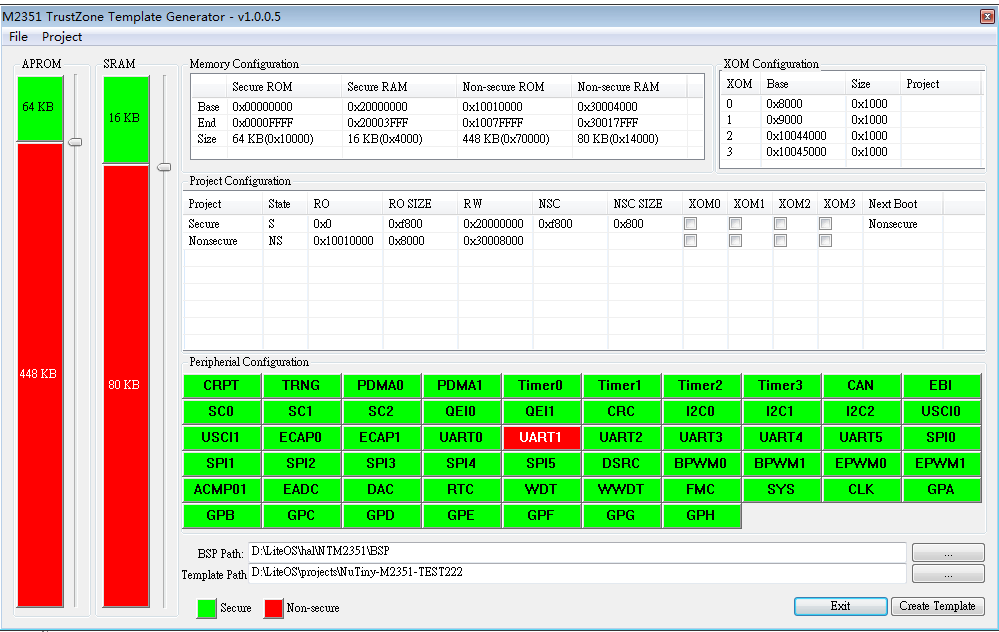


图2-1

## 工程说明

创建的工程有两个，一个工程运行于安全区，另一个工程运行于非安全区。Project->options for target->Target中可见设置（图2-2）

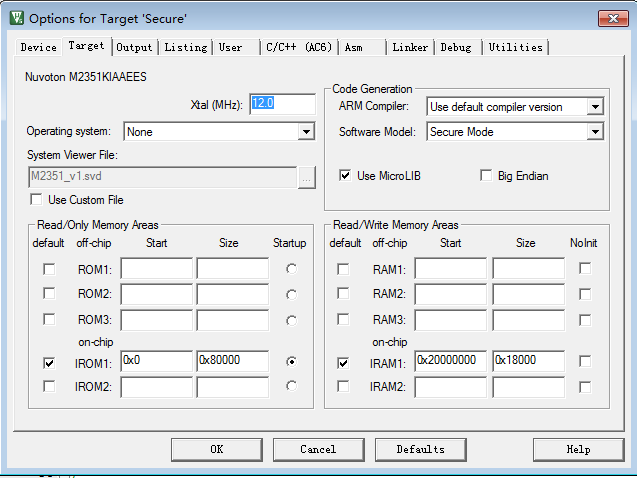


图2-2

## TrustZone程序说明

程序从secure工程开始运行，secure工程中对时钟外设等进行初始化设置，提供可供Non\_secure调用的APIs，其中外设选用secure还是Non\_secure可以用partition\_M2351.h（该文件在secure工程中）中的configuration Wizard功能设置（如图2-3），secure区可以控制所有设置为secure或Non\_secure的外设。Secure区运行完后会运行Boot\_Init()函数跳转到Non\_secure区工程，用户函数主要跑在Non\_secure区，Non\_secure区可以控制设置为Non\_secure的各种外设，如果要想控制secure外设，只能通过调用secure区提供的API。

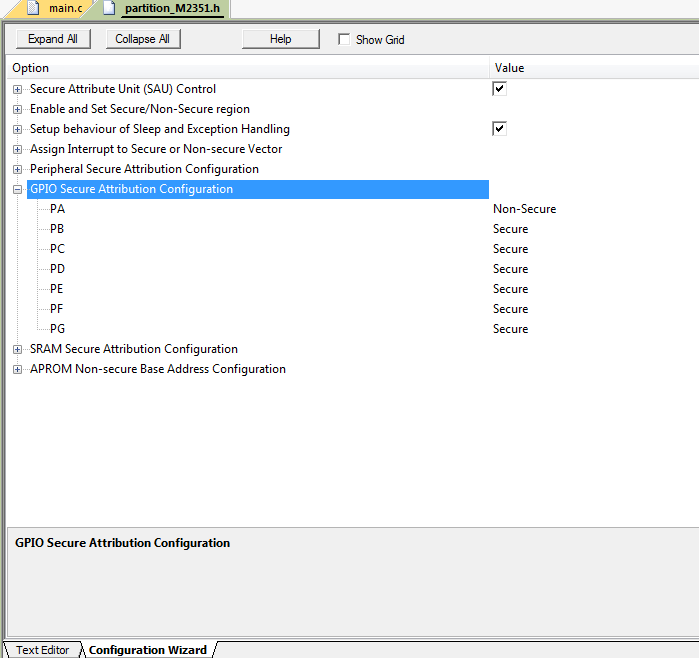


图2-3

## LiteOS移植

由于用户程序主要跑在非安全区，所以LiteOS系统也是移植在非安全工程，非安全区除了控制安全外设需要调用安全区提供的API外，其他程序编写等同普通cpu。

## 程序烧录步骤，

non-secure工程先build、download，secure工程后build，然后download secure code直接运行。