

Cortex-M 対応 MULCOS-MK

(MULti Core Operating System – Multilayer Kernel)

NUCLEO-H536ZI サンプルガイド



第1版 2025年4月

改訂履歴

版数	発行日	改訂内容
第 1 版	2025 年 4 月	Cortex-M 対応 MULCOS-MK μ ITRON4.0 仕様版 NUCLEO-H536ZI サンプルガイドの初版を発行

はじめに

MULCOS-MK (MULTi Core Operating System - Multilayer Kernel) とは、セキュリティレベルの異なる階層化したリアルタイム・オペレーティング・システム (Real Time Operating System、以降 RTOS と省略) の実行環境を管理する多層化カーネル (以降では、本製品を単に本カーネルと呼ぶ) です。

μ ITRON4.0 仕様版の RTOS (以降では、単に本 RTOS と呼ぶ) は、最も多く使われている仕様の 1 つで、当時の社団法人トロン協会が策定し公開した「 μ ITRON4.0 仕様」(現在は、トロンフォーラムが公開) に準拠しています。

本書は ST マイクロエレクトロニクス NUCLEO-H536ZI 評価ボードを使用したサンプルプログラムの説明書です。本書以外の説明書として、Cortex-M 対応 MULCOS-MK の μ ITRON4.0 仕様版共通部ユーザーズガイドと、ST デバイス依存部ユーザーズガイドとが用意されています。

(μ ITRON とは Micro Industrial TRON、TRON とは The Realtime Operating system Nucleus の略称)

参考資料

- ・社団法人トロン協会発行「 μ ITRON4.0 仕様 Ver.4.03.00」
- ・Arm Limited 社発行「Arm[®]v8-M Architecture Reference Manual」

目次

改訂履歴	0
はじめに	1
目次	2
第 1 章 環境の説明	3
1-1. ハードウェア設定	3
1-2. フォルダ構成	6
1-3. プロジェクトファイル	7
第 2 章 サンプルプログラムの説明と動作	8
2-1. メモリマップ	8
2-2. サンプルの初期化	9
2-2-1. メモリマップに依存した初期化	9
2-2-2. クロックの初期化	9
2-2-3. デバイスとポートのセキュア属性の初期化	10
2-2-4. 割込み要因のセキュア属性の初期化	10
2-3. サンプルプログラムの動作	11
2-3-1. セキュア層の処理	11
2-3-2. 非セキュア層の処理	11
第 3 章 サンプルプログラムの起動	12
3-1. サンプルプログラムのデバッグ	12

第 1 章 環境の説明

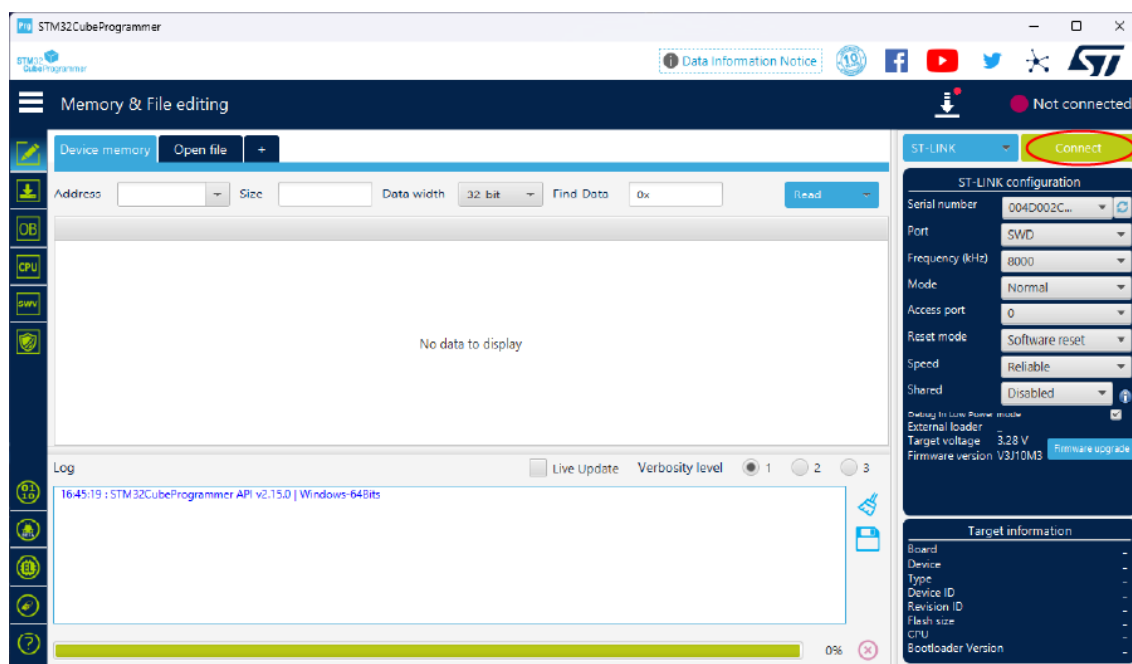
Cortex-M 対応 MULCOS-MK の NUCLEO-H536ZI 評価ボードのサンプルプログラムのプロジェクトでは、以下のボードと開発ツールを使用しています。

- [NUCLEO-H536ZI STM32 Nucleo-144 boards](#)
- [STM32CubeIDE STM32 用統合開発環境](#)
- [STM32CubeProg STM32CubeProgrammer software for all STM32](#)

1-1. ハードウェア設定

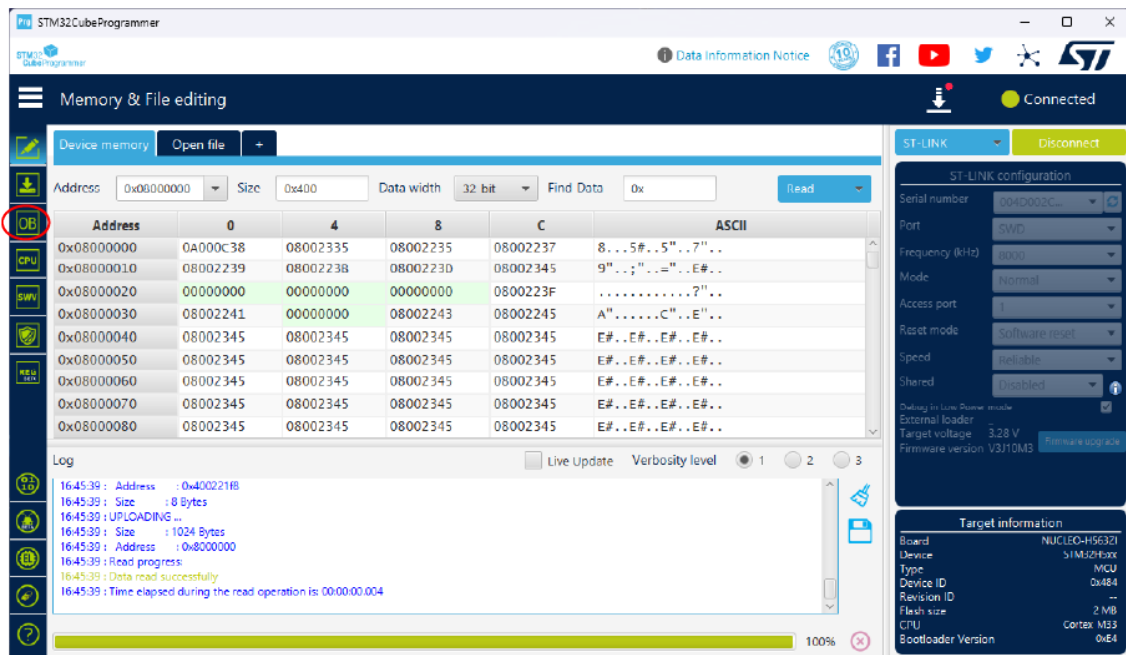
プロジェクトは、NUCLEO-H536ZI 評価ボードのオンボードデバッガ（CN1）を接続し、UART3 が仮想 UART 経由で通信します。

NUCLEO-H536ZI 評価ボードの初期状態では TrustZone が無効になっているため、オプションバイトを STM32CubeProg を使用して書き換えます。

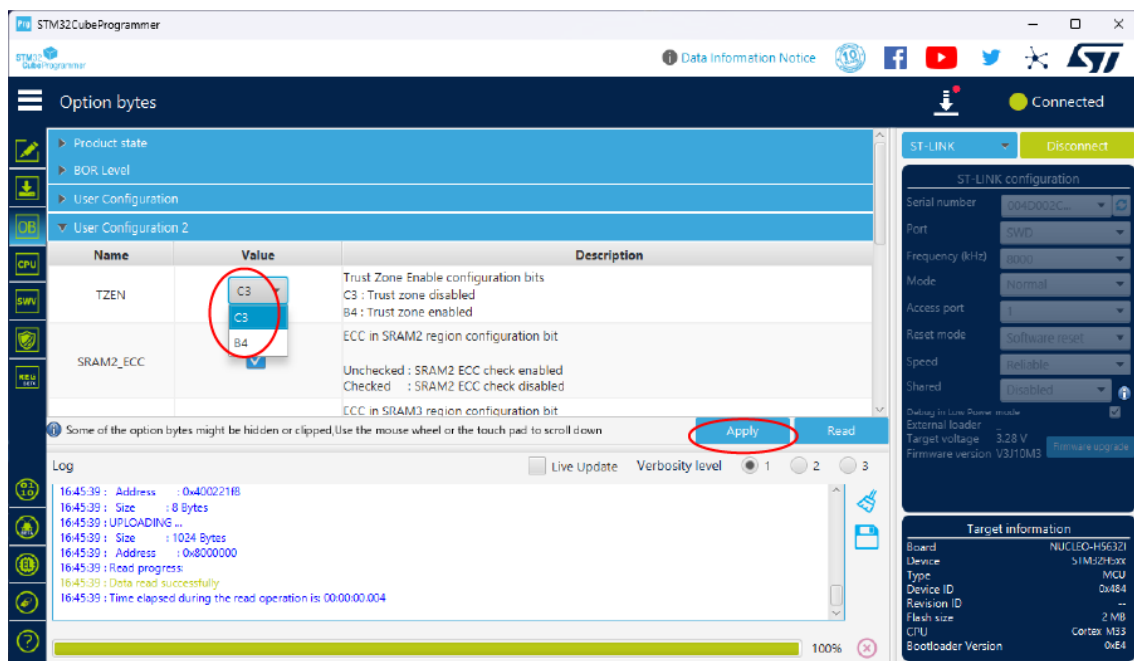


STM32CubeProg を起動し、「Connect」をクリックして、NUCLEO-H536ZI 評価ボードと接続します。

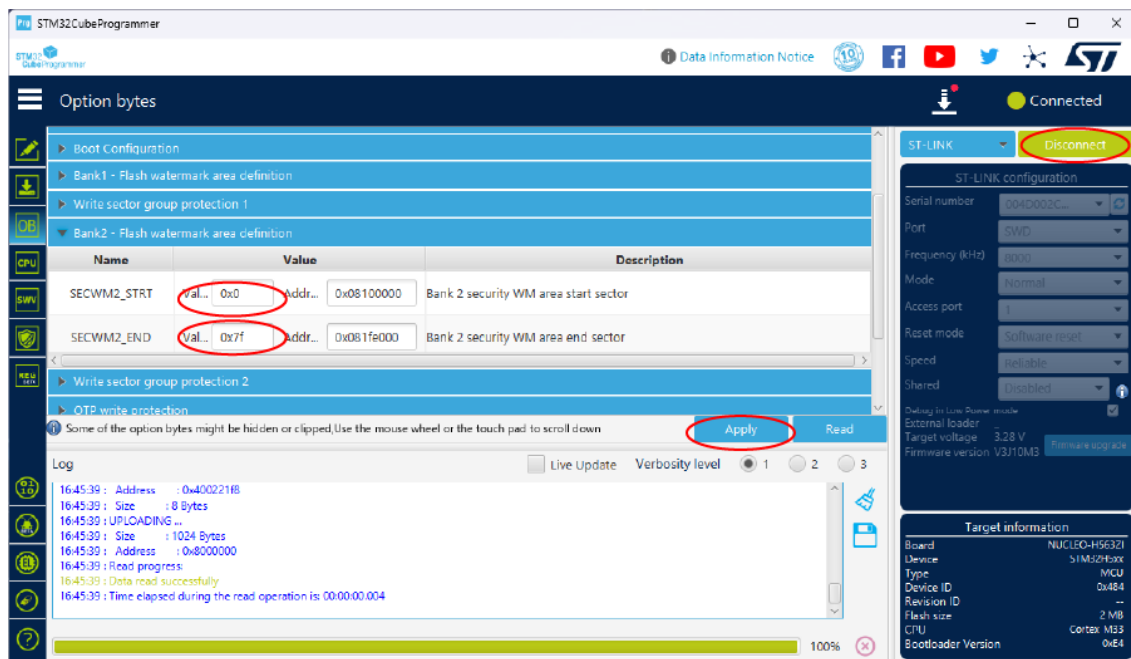
「OB」をクリックし、オプションバイトを選択します。



次に、「User Configuration 2」をクリックします。「TZEN」の初期状態では TrustZone の無効を表す「C3」になっており、これを有効を表す「B4」に変更した後、「Apply」をクリックし更新します。



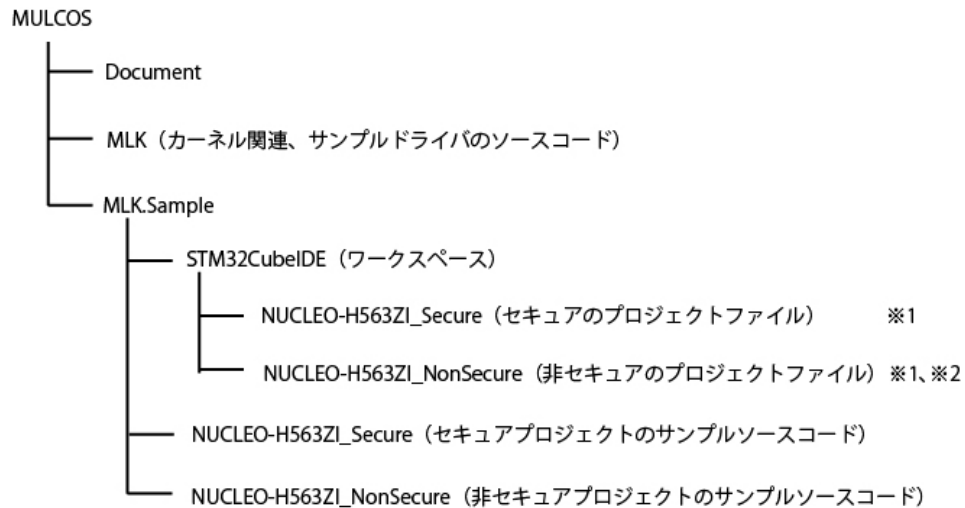
そして、「Bank2 – Flash watermark area definition」をクリックします。フラッシュメモリのバンク 2 は、セクション 0 から 127 まだがセキュア領域に設定されています。これの全領域を非セキュア領域にするため、SECWM2_STRT > SECWM2_END の関係が成り立つよう、ここでは SECWM2_STRT を 0x00 から 0x7f に、SECWM2_END を 0x7f から 0x00 に変更した後、「Apply」をクリックし更新します。



最後に、「Disconnect」をクリックして終了します。

1-2. フォルダ構成

サンプルプログラムに関連するフォルダは、以下のファイル構成でビルドされています。また、プロジェクトファイルは、ビルドの対象ファイルをコピーせずにリンクすることもできますが、一時的にソース修正することができるように、ファイルはコピーしています。

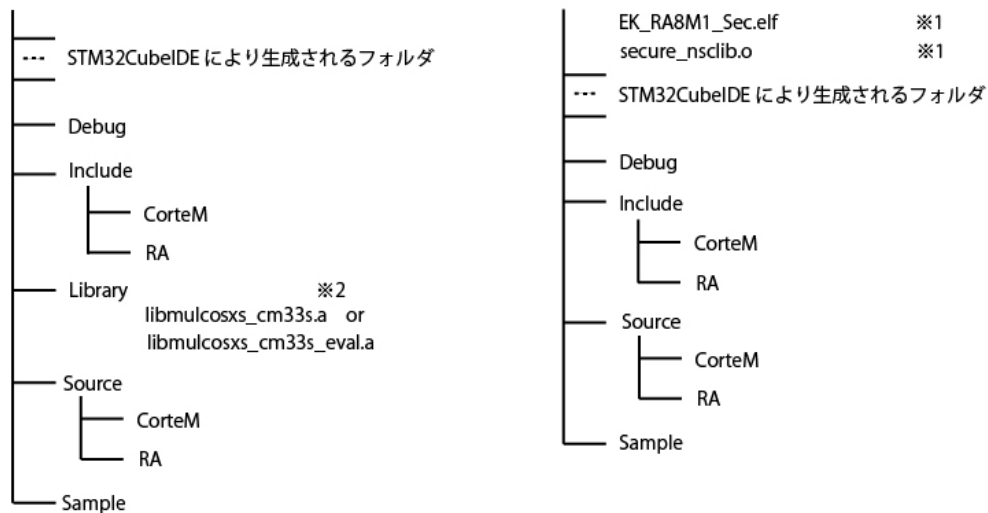


タイプ 2 評価版では両セキュアのプロジェクトファイル (※1)、タイプ 1 評価版では非セキュアのプロジェクトファイル (※2) のみの配布となります。

1-3. プロジェクトファイル

サンプルプログラムは、セキュアと非セキュアの 2 つのプロジェクトで構成され、評価版はこれらがエクスポートされています。

NUCLEO-H563ZI_Secure (セキュアのプロジェクトファイル) NUCLEO-H563ZI_NonSecure (非セキュアのプロジェクトファイル)



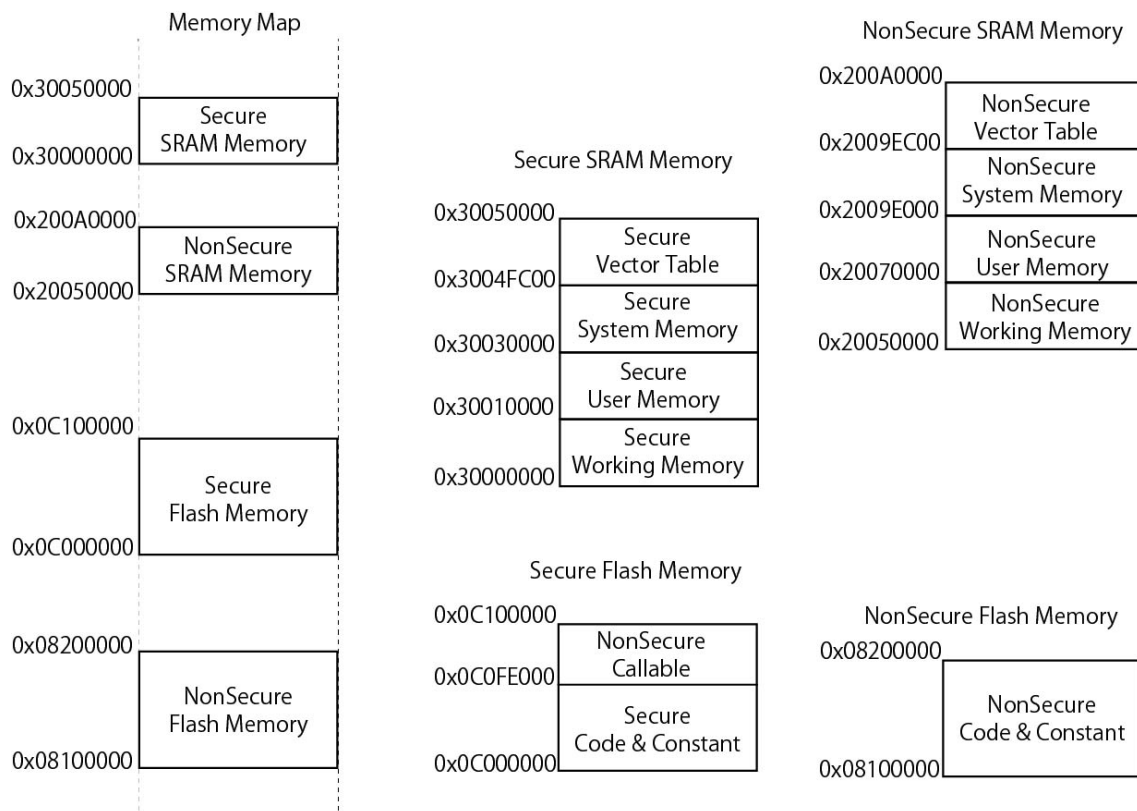
タイプ 1 評価版では非セキュアのプロジェクトファイルのみですが、セキュアのフラッシュメモリへの書き込みに使用される NUCLEO-H563ZI_Secure.elf ファイル (※1) と非セキュアのビルドに必要な NSC ライブラリ secure_nsclib.o (※1) を追加しています。Library フォルダ (※2) には、標準版ではカーネルライブラリ (libmulcosxe_cm85s.a) が、タイプ 2 評価版では制限版のカーネルライブラリ (libmulcosxe_cm85s_eval.a) が含まれています。

第 2 章 サンプルプログラムの説明と動作

NUCLEO-H536ZI 評価ボード用サンプルプログラムのプロジェクトについて説明します。

2-1. メモリマップ

サンプルプログラムでは、メモリ領域を使用状況に合わせて、以下のように定義しています。



RTOS のみが読み書きする特権モードでのみアクセス可能な領域として、

Vector Table 例外ベクタテーブル

System Memory RTOS の制御データとメインスタック領域

アプリケーションを含めて読み書きし、特権モードに限らずアクセス可能な領域として、

User Memory タスクのスタック領域とメモリプール領域

Working Memory アプリケーションが確保し利用する領域

RTOS やアプリケーションのコードと定数の読み出しのみ領域として、

Code & Constant 基本的にコードは RTOS とアプリケーションとを混同した領域

2-2. サンプルの初期化

SAU や MPU のメモリマップに依存する初期化の他に、デバイスとポートの初期化があります。デバイスとポートでは、タイプ 1 評価版を考慮し、セキュア層で使用する UART と LED 以外は非セキュアで利用できる設定にしています。通常的设计では、それぞれのデバイスやポートをセキュア層で使用するのか、非セキュア層で使用するのかを設計時に決定します。

- メモリマップに依存した初期化
- クロックの初期化
- デバイスとポートのセキュア属性の初期化
- 割込み要因のセキュア属性の初期化

2-2-1. メモリマップに依存した初期化

メモリマップに対応した値で、SAU と MPU は初期化されています。

2-2-2. クロックの初期化

発振器と PLL の初期化により、周波数は以下の設定にしています。

クロック名	周波数
CPUCLK	250MHz
PCLK1P	250MHz
PCLK1Q	Disable
PCLK1R	Disable

以下の設定で、非セキュアや非特権で保護していないため、未設定クロックを使用することは可能です。

レジスタ名	値
RCC_SECCFGR	0x00000000
RCC_PRIVCFGR	0x00000000

2-2-3. デバイスとポートのセキュア属性の初期化

サンプルプログラムでは、セキュア層で使用している UART9 と LED1-3 に関するデバイスやポート以外では非セキュア属性に、すべて非特権アクセスに初期化しています。

レジスタ名	値
GTZC1_TZSC_SECCFGR1	0x00004000
GTZC1_TZSC_SECCFGR2	0x00000000
GTZC1_TZSC_SECCFGR3	0x00000000
GTZC1_TZSC_PRIVCFGR1	0x00000000
GTZC1_TZSC_PRIVCFGR2	0x00000000
GTZC1_TZSC_PRIVCFGR3	0x00000000
GPIOA_SECCFGR	0x00000000
GPIOB_SECCFGR	0x00000001
GPIOC_SECCFGR	0x00000000
GPIOD_SECCFGR	0x00000300
GPIOE_SECCFGR	0x00000000
GPIOF_SECCFGR	0x00000010
GPIOG_SECCFGR	0x00000010
GPIOH_SECCFGR	0x00000000
GPIOI_SECCFGR	0x00000000

2-2-4. 割り込み要因のセキュア属性の初期化

割り込みで使用する例外では、UART3 以外を非セキュアで使用できるように初期化しています。

レジスタ名	値
NVIC_ITNS0	0x00004000
NVIC_ITNS1	0x00000000
NVIC_ITNS2	0x00000000
NVIC_ITNS2	0x00000000

2-3. サンプルプログラムの動作

サンプルプログラムはセキュア層と非セキュア層とで、2つのプロジェクトファイルに分割されており、UARTデバイスとLEDがセキュア層に配置すべき処理だと仮定しています。

セキュア層では以下の関数をセキュアライブラリとして定義しています。

関数名	説明
void Led1_On(void);	ユーザ LED1 の点灯
void Led1_Off(void);	ユーザ LED1 の消灯
void Led2_On(void);	ユーザ LED2 の点灯
void Led2_Off(void);	ユーザ LED2 の消灯
void Led3_On(void);	ユーザ LED3 の点灯
void Led3_Off(void);	ユーザ LED3 の消灯
ER ini_uart3(T_UART_DEF *uart_def);	UART3 の初期化
ER ref_uart3(T_UART_REF *uart_ref);	UART3 の状態参照
ER ctr_uart3(UINT cdata);	UART3 の制御
ER fls_uart3(TMO timeout);	UART3 の送待待ち
ER putc_uart3(VB data, TMO timeout);	UART3 の一文字送信
ER puts_uart3(VB *data, UINT *count, TMO timeout);	UART3 の文字列送信
ER getc_uart3(VB *data, UB *sts, TMO timeout);	UART3 の一文字受信
ER gets_uart3(VB *data, UB *sts, UINT *count, TMO timeout);	UART3 の文字列受信

2-3-1. セキュア層の処理

UART3 を下記で初期化し、セキュアの起動メッセージを表示後は以下の処理を行います。

ボーレート	115200BPS
パリティ	パリティなし
ストップビット	1

- ユーザ LED3 を 500msec 間隔で 100msec 点灯

2-3-2. 非セキュア層の処理

UART3 より、非セキュアの起動メッセージを表示後は以下の処理を行います。

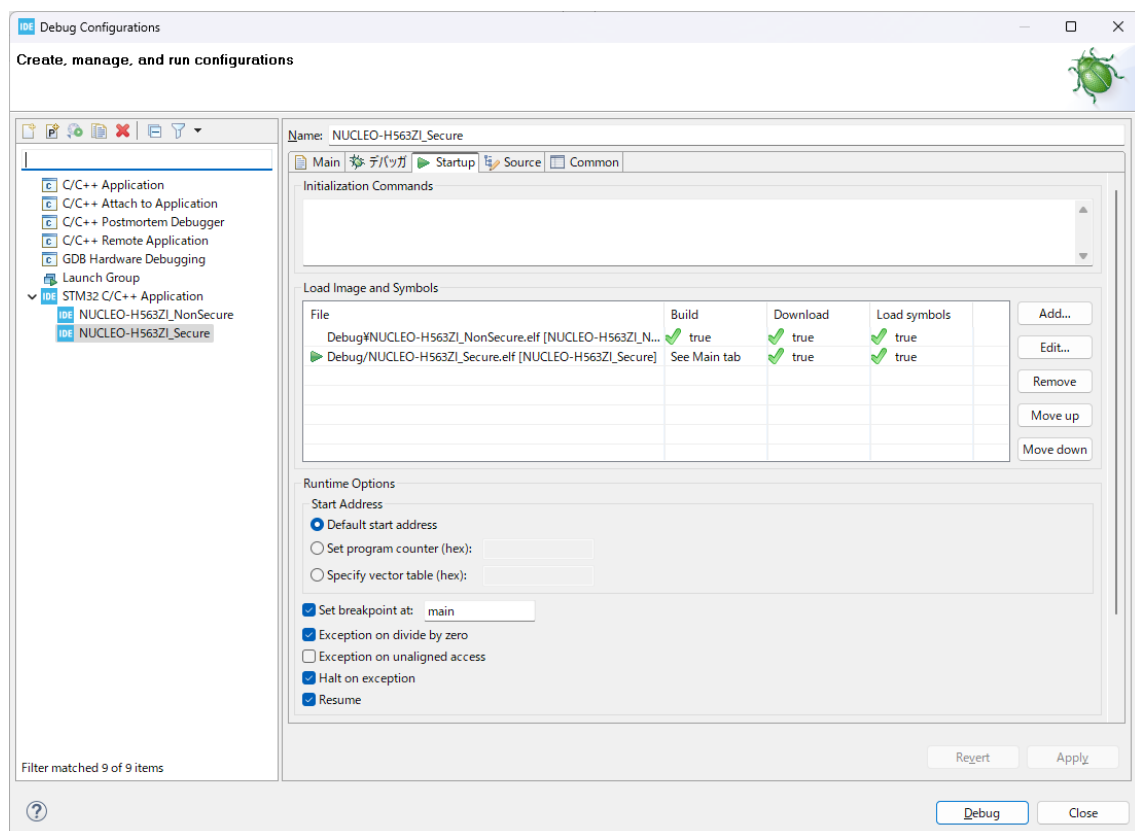
- ユーザ LED1 を 500msec 間隔で 50msec 点灯
- ユーザボタンの押下でメッセージを表示し、LED2 を 2000msec 点灯

第3章 サンプルプログラムの起動

NUCLEO-H536ZI 評価ボード用サンプルプログラムの起動について説明します。また、使用されるサンプルプロジェクトは、インポートを終えてください。

3-1. サンプルプログラムのデバッグ

STM32CubeIDE を起動し、メニュー「Run」→「Debug configuration」により「Create, manage, and run configurations」画面を表示し、「STM32 C/C++ Application」→「NUCLEO-H536ZI_Secure」を選択し「Startup」タブをクリックします。タイプ1評価版ではセキュアプロジェクトが存在しないため、「NUCLEO-H536ZI_NonSecure」を選択します。



初期設定では、セキュアのelfファイルと非セキュアのelfファイルをロードするようになっており、この状態で「Debug」をクリックすると実行イメージをロードしデバッガが起動します。「Set breakpoint at:」で main を指定している場合、main0関数でブレークし、「Resume」のクリックでプログラムは再開します。

この main0関数は、セキュア層と非セキュア層とに存在しますが、処理順序でセキュア層の main0関数でブレークします。ただし、セキュア層のelfファイルのロードタイプに「イ

メージのみ」を選択すると非セキュア層の `main()`関数でブレークします。また、タイプ 1 評価版にはセキュア層のソースコードが存在しないため、非セキュア層の `main()`関数でブレークします。

タイプ 1 評価版では、セキュア層のソースコードを含まないため、セキュア層の elf ファイルは変更されることがありません。そのため、他のプログラムをロードした後の場合のみ、セキュア層の elf ファイルをロードすれば良く、以降はファイル名の左のチェックボックスを外しロードをスキップすることができます。

(余白)

(余白)

Cortex-M 対応 MULCOS-MK

NUCLEO-H536ZI サンプルガイド

2025 年 4 月 第 1 版発行

株式会社スパイラルテック

Mail tech-info@spiral-tech.co.jp