life.augmented

マイコン初心者にも簡単！ 高性能STM32H7シリーズ入門

STマイクロエレクトロニクス株式会社
マイクロコントローラ & デジタル製品グループ
マイクロコントローラ製品技術部 部長

原 文雄

拡大し続けるSTM32ポートフォリオ

マイクロプロセッサ

STM32MP1
Up to 1 GHz Cortex-A7
209 MHz Cortex-M4

STM32MP2
Dual 1.5 GHz Cortex-A35
400 MHz Cortex-M33

ハイパフォーマンスマイコン

STM32F7
1082 CoreMark
216 MHz Cortex-M7

STM32H7
Up to 3224 CoreMark
Up to 550 MHz Cortex -M7
240 MHz Cortex -M4

STM32N6
MCU with neural processing unit

STM32F2
Up to 398 CoreMark
120 MHz Cortex-M3

STM32F4
Up to 608 CoreMark
180 MHz Cortex-M4

STM32H5
Up to 1023 CoreMark
250 MHz Cortex-M33

メインストリームマイコン

STM32F3
245 CoreMark
72 MHz Cortex-M4

STM32G4
569 CoreMark
170 MHz Cortex-M4

ミックスド・シグナル・マイコン

STM32C0
114 CoreMark
48 MHz Cortex M0+

STM32F0
106 CoreMark
48 MHz Cortex-M0

STM32G0
142 CoreMark
64 MHz Cortex-M0+

STM32F1
177 CoreMark
72 MHz Cortex-M3

超低消費電力マイコン

STM32L0
75 CoreMark
32 MHz Cortex-M0+

STM32U0
118 CoreMark
48 MHz Cortex M0+

STM32L4
273 CoreMark
80 MHz Cortex-M4

STM32L4+
409 CoreMark
120 MHz Cortex-M4

STM32L5
443 CoreMark
110 MHz Cortex-M33

STM32U5
651 CoreMark
160 MHz Cortex-M33

ワイヤレスマイコン

STM32WL
162 CoreMark
48 MHz Cortex-M4
48 MHz Cortex-M0+

STM32WB
216 CoreMark
64 MHz Cortex-M4
32 MHz Cortex-M0+

STM32WBA
407 CoreMark
100 MHz Cortex-M33

直近に量産移行した製品　2024年発表の新シリーズ　開発中

life.augmented

STM32H7シリーズの特徴

STM32マイコンのポートフォリオを拡張する新製品

シリーズ最高性能の実現

- デュアルコア品　最大 2424 + 800 CoreMark
  （Arm®Cortex®-M7 @480MHz + Cortex-M4 @240MHz）
- シングルコア品　最大 3174 CoreMark
  （Cortex-M7 @600MHz）

柔軟なシングル / デュアルコア·アーキテクチャ

産業·セキュリティ·AIアプリケーションに最適

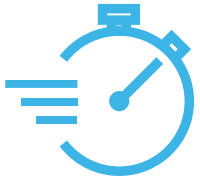
- グラフィック制御アシスト
- 高速データ転送
- 高度な周辺機能

高度なセキュリティ機能

- 暗号ハッシュ
- Cortex-M7セキュリティ·サービス

開発を加速する豊富なエコシステム

- 各種SWツール
- 評価用ボード
- コミュニティ & 豊富なパートナー



life.augmented

170品種以上をラインアップ！
STM32H7シリーズ 製品ポートフォリオ

ブート Flash	STM32**H7R3/7S3**	STM32**H7R7/7S7**				
	600MHz / 1284DMIPS	600MHz / 1284 DMIPS				
	SRAM 620KB 64K user flash 128K ST-iRoT	SRAM 620KB 64K user flash 128K ST-iRoT				
	Chrom-ART	NeoChrom				
デュアルコア					STM32**H745/755**	STM32**H747/757**
					480 + 240MHz 1027 + 300DMIPS	480 + 240MHz 1027 + 300DMIPS
					RAM 1 MB	RAM 1 MB
					Flash up to 2 MB	Flash up to 2 MB
シングルコア	STM32**H7A3/B3**	STM32**H742**	STM32**H743/753**	STM32**H725/735**	STM32**H723/733**	
	280MHz / 599DMIPS	480MHz / 1027DMIPS	480MHz / 1027DMIPS	550MHz / 1177DMIPS	550MHz / 1777DMIPS	
	RAM 1.4MB	RAM 692 KB	RAM 1 MB	RAM 564KB	RAM 564 KB	
	Flash up to 2MB	Flash up to 2 MB	Flash up to 2 MB	Flash up to 1MB	Flash up to 1 MB	
$ バリューライン	STM32**H7B0**	STM32**H750**	STM32**H730**			
	280MHz / 599DMIPS	480MHz / 1027DMIPS	550MHz / 1177DMIPS			
	RAM 1.4MB	RAM 1 MB	RAM 564 BB			
	Flash 128KB	Flash 128 KB	Flash 128 KB			

Arm® Cortex® **-M7** **-M7 & -M4**

life.augmented

STM32H723 / H733
汎用ライン

豊富な周辺機能、高性能、大容量メモリ、3つのパワー·ドメイン、LDO、STM32H753 / H743ラインとのパッケージ互換

- パフォーマンス
- RAM配置
- ECC（メモリ全領域）
- 複数の高速 A/DC
- グラフィック
- セキュリティ
- 数値演算アクセラレータ
- メモリ·インタフェース
- 通信インタフェース
- 電源供給スキーム
- STM32 DNA（互換性を考慮、パッケージ、エコシステム等）

life.augmented

STM32H723 / H733
汎用ライン

（*）：STM32H733 でのサポート（暗号機能あり）

life.augmented

STM32開発エコシステム

ソフトウェア

ソフトウェア ツール

STM32 CubeMX
構成

STM32 CubeIDE
開発

STM32 CubeProgrammer
プログラム

STM32 CubeMonitor
モニタリング

組込み ソフトウェア

STM32 CubeMCU Packages
パッケージ

STM32 Cube

拡張
STM32 CubeExpansion

ハードウェア

STM32 Nucleo-144ボード

STM32 Nucleo

ディスカバリ·キット

カスタマーサポート

FAE - Worldwide
カスタマ·サポート

community.st.com

MOOC

Partner Program

life.augmented

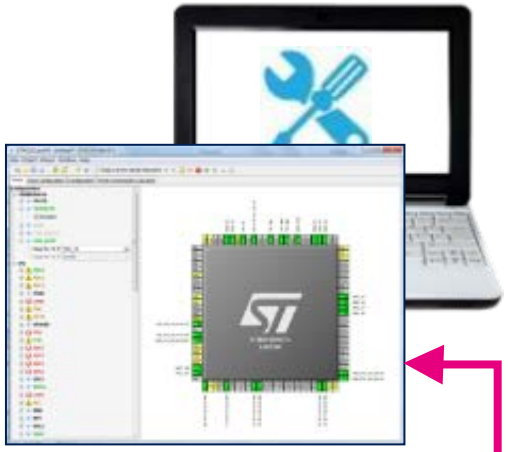
STM32CubeMX
組込みソフトウェア開発支援ツール

life.augmented

組込みソフトウェア開発支援ツール STM32CubeMX

STM32CubeMXを使用した開発形態

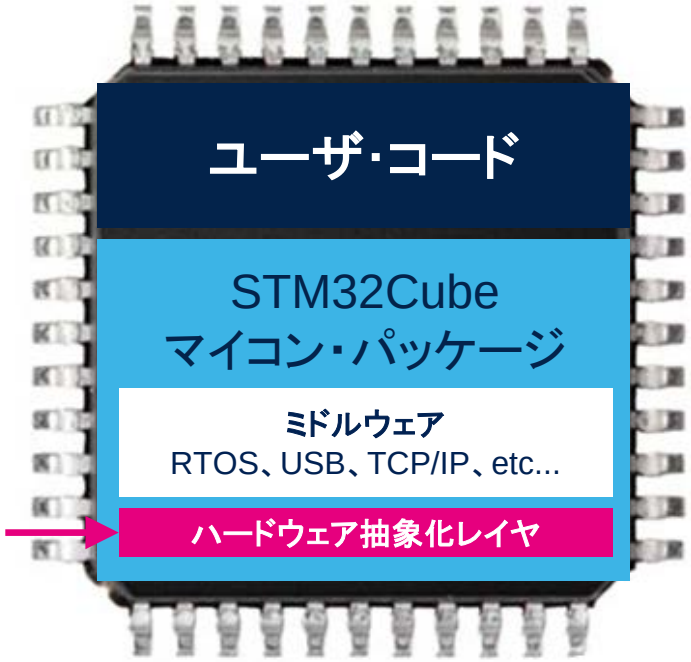
STM32Cube**MX**
（PC上の設定ツール）



ユーザの設定に応じた
Cコードの初期化ファイルを
自動生成

STM32製品間の差分を吸収する
抽象化レイヤ（HAL）

ピン･アサイン･ツールで各ペリフェラルを簡単設定 & コード生成



- **ペリフェラル、クロック、割り込み、DMA等の設定コードを自動生成するツール**
- **ユーザの開発負荷を削減 & STM32間のソフトウェア移植性を実現**

life.augmented

STM32CubeMX 開発支援ツール

さまざまなマイコン設定をGUIで簡単制御
初期プロジェクトを生成し、組込み製品開発の期間短縮に貢献

消費電力計算

最新の設計リソースへ簡単アクセス
最適なマイコン製品の検索・比較

ピン・アサインメントの設定
既存プロジェクトからの移行

クロック・ツリーの初期化

ペリフェラル設定

ソフトウェア・コンポーネント

セキュリティ
ブート・シーケンスの管理

初期化コード生成

IAR SYSTEMS
arm KEIL
STM32 CubeIDE
CODE
Azure RTOS
CAN
I2C
SPI
UART

life.augmented

Our technology starts with You

Find out more at STM32H7

life.augmented

life.augmented

NUCLEO-H723ZG +CubeIDE プログラミング

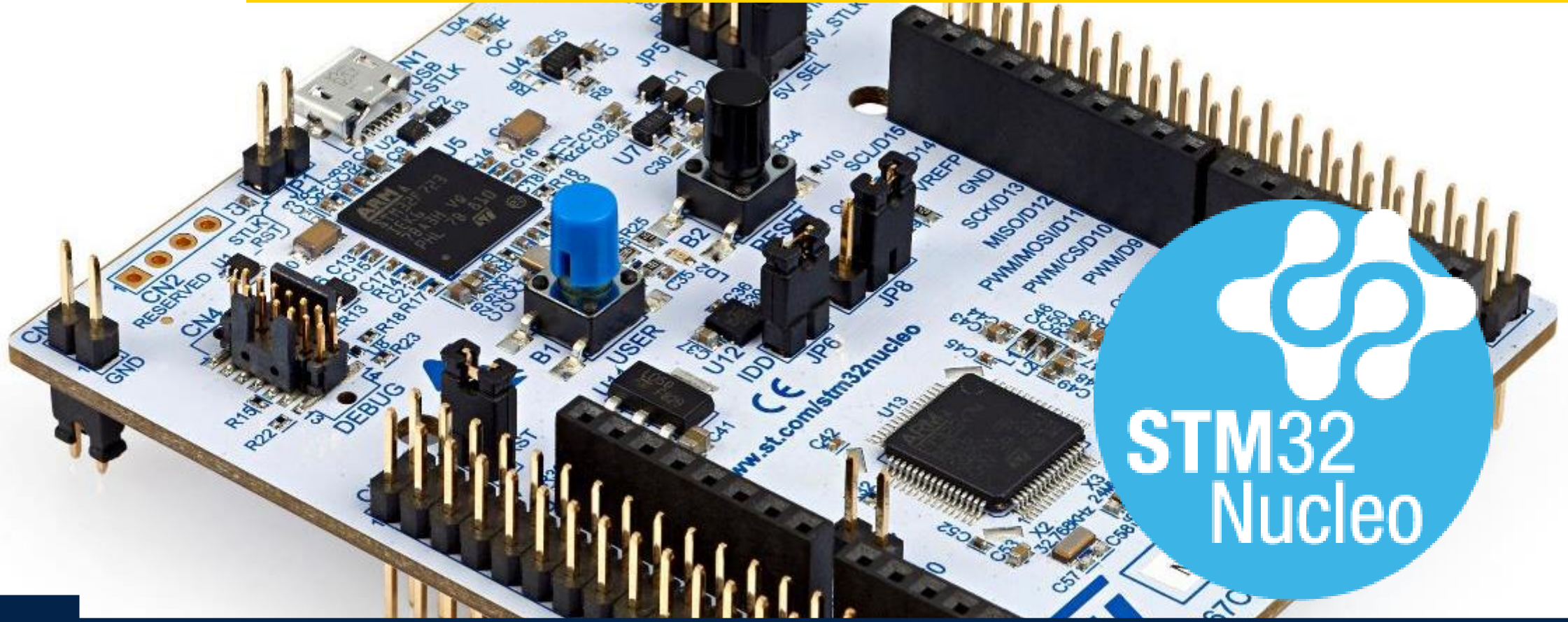
2024年05月10日

STマイクロエレクトロニクス株式会社
マイクロコントローラ製品技術部

STM32

STM32 マイコン開発ボード (NUCLEO)

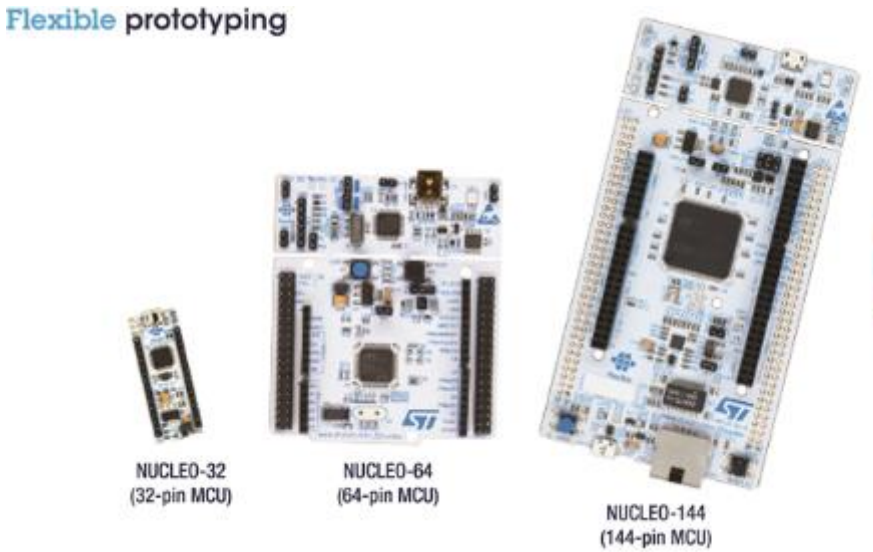
STM32マイコン搭載プロトタイプ作成用
オープン開発プラットフォーム



life.augmented

STM32 Nucleoボード 概要

特　徴



- **32 / 64 / 144ピン･パッケージのSTM32マイコン搭載**
- **ST-LINK/V3 デバッガ / プログラマ搭載**
  - 仮想のcomポート（Virtual com port）
  - マス･ストレージ
- **専用シールドによる広範な拡張機能**
  - Arduino Uno Rev3接続に対応（Nucleo-64 / Nucleo-144）
  - ST Zioコネクタを介して幅広いペリフェラルにアクセス可能（Nucleo-144）
  - ST Morphoコネクタを介してマイコンの全端子にアクセス可能（Nucleo-64 / Nucleo-144）
  - Arduino-Nanoコネクタ（Nucleo-32）
- **mbedオンライン･リソースへの直接アクセス**
- **IAR、Keil、arm mbedオンライン、GCCベースのIDEによるサポート**

life.augmented

全72種類

STM32 Nucleoボードのラインアップ
（2024年4月時点）

ハイパフォーマンスマイコン		STM32**F2** 144: F207ZG*	STM32**F4** 64: F401RE, F410RB, 64: F411RE, F446RE 144: F412ZG, F413ZH 144: F439ZI* 144: F446ZE	STM32**H5** 64: H503RB 64: H533RE 144: H563ZI NEW	STM32**F7** 144: F722ZE 144: F756ZG*, F767ZI*	STM32**H7** 144: H753ZI* 144: H723ZG*, H7A3ZI-Q 144: H755ZI-Q*, 255: H7S3L8* NEW
メインストリームマイコン	STM32**C0** 64(48): C031C6	STM32**F0** 32: F031K6, F042K6 64: F030R8, F070RB 64: F072RB, F091RC	STM32**G0** 32: G031K8 64: G070RB, G071RB 64: G0B1RE	STM32**F1** 64: F103RB	STM32**F3** 64: F302R8, F303K8 64: F303RE, F334R8 144: F303ZE	STM32**G4** 32: G431KB 64: G431RB, G474RE 64: G491RE
超低消費電力マイコン	STM32**L0** 32: L011K4, L031K6 64: L010RB, L053R8 64: L073RZ STM32**L1** 64: L152RE	STM32**U0** 64: U031R8, U083RC NEW	STM32**L4** 32: L412KB, L432KC 64: L452RE, L476RG 64: L412RB-P, L433RC-P 64: L452RE-P 144: L496ZG*, L4A6ZG 144: L496ZG-P*	STM32**L4+** 144: L4P5ZG, L4R5ZI 144: L4R5ZI-P	STM32**L5** 144: L552ZE-Q*	STM32**U5** 144: U575ZI-Q 144: U5A5ZJ-Q 64: U545RE-Q
ワイヤレスマイコン			STM32**WL** 64: WL55JC	STM32**WB** WB55 48: WB15CC 64: WB55RG	STM32**WBA** 64: WBA52CG, WBA55CG NEW	

マセンダ：SMPS搭載（-P: 外部SMPS、-Q: 内蔵SMPS）

*: イーサネット：10/100Mbps

USBドングル付き68ピン

HW暗号化 / ハッシュ・プロセッサ搭載

新ボード　USB Type-C® シンク　USB-PD

life.augmented

ツールの説明

開発環境

評価ボード

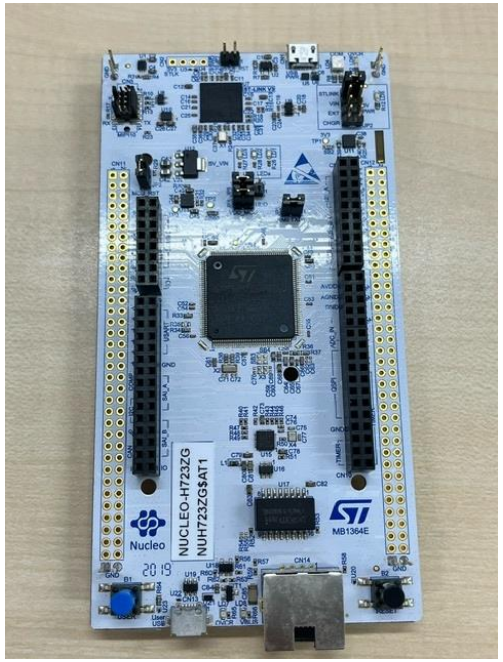
- NUCLEO-H723ZG

統合開発環境

- ST社　STM32CUBEIDE

ソフトウェア･ライブラリ

- STM32CubeMX
- STM32CubeH7



CMSIS COMPLIANT

事前準備

確認事項

1. NUCLEO-H723ZG

- URL: https://www.stmcu.jp/design/hwdevelop/nucleo/77343/
- 関連ドキュメント
  - UM2407 STM32H7 Nucleo-144 ボード（MB1364） ユーザー マニュアル
  - URL: https://www.stmcu.jp/design/document/users_manual/65701/

2. STM32CubeIDE

- URL: https://www.stmcu.jp/design/sw_dev/pc_soft/66469/
- 関連ドキュメント
  - UM2609 STM32CubeIDE ユーザ･ガイド（和訳版あり）
  - URL: https://www.stmcu.jp/design/document/users_manual/76085/
- Wiki Page
  - URL:https://wiki.st.com/stm32mcu/wiki/STM32CubeIDE:Introduction_to_STM32CubeIDE

注意！
CubeIDEは管理者権限でインストールしましょう。

life.augmented

事前準備

確認事項

3. STM32CubeMX

- URL: https://www.stmcu.jp/design/sw_dev/pc_soft/52798/
- 関連ドキュメント
  - UM1718 STM32CubeMX：STM32の設定と初期化のCコード生成
  - URL:https://www.stmcu.jp/design/document/users_manual/53167/
- Wiki Page
  - https://wiki.st.com/stm32mcu/wiki/Introduction_to_STM32CubeMX

注意！
CubeMXは管理者権限でインストールしましょう。

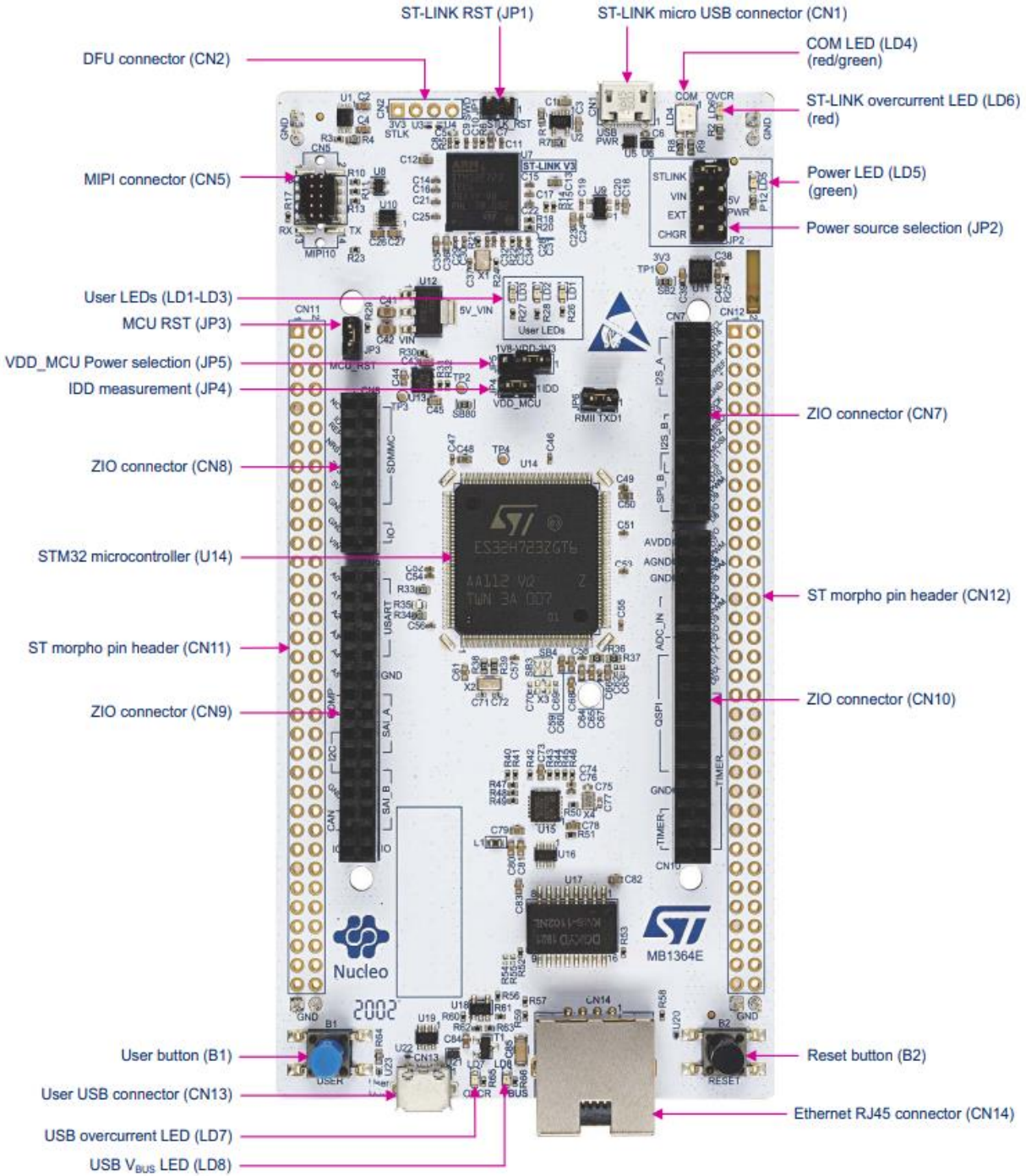
4. STM32CubeH7

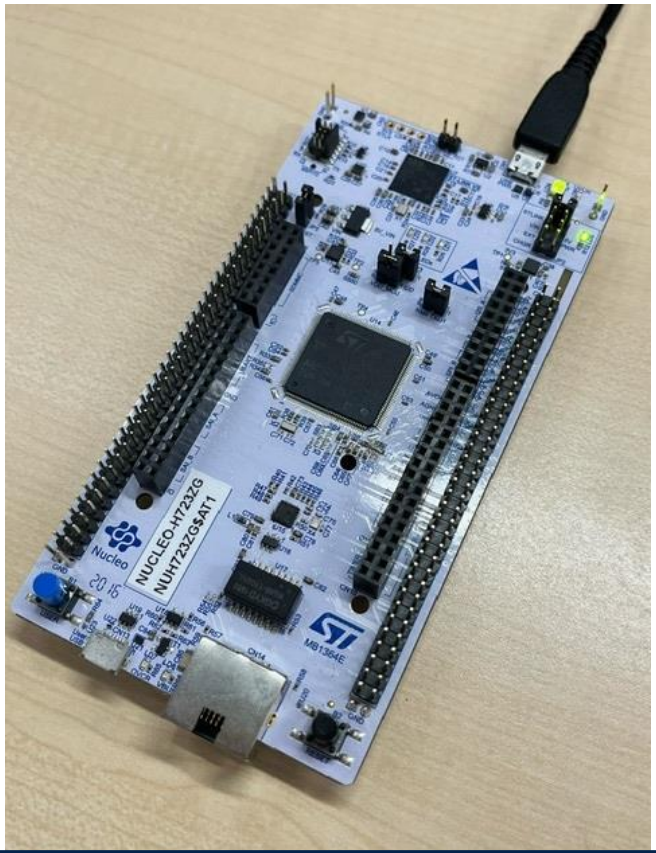
- URL: https://www.stmcu.jp/design/sw_dev/firmware/53422/
- 関連ドキュメント
  - UM2217 Description of STM32H7 HAL and low-layer drivers
  - URL: https://www.stmcu.jp/design/document/users_manual/53956/

注意！
CubeH7のインストール時には日本語表記のディレクトリ名はなるべく控えましょう。

life.augmented

Nucleo ボード説明

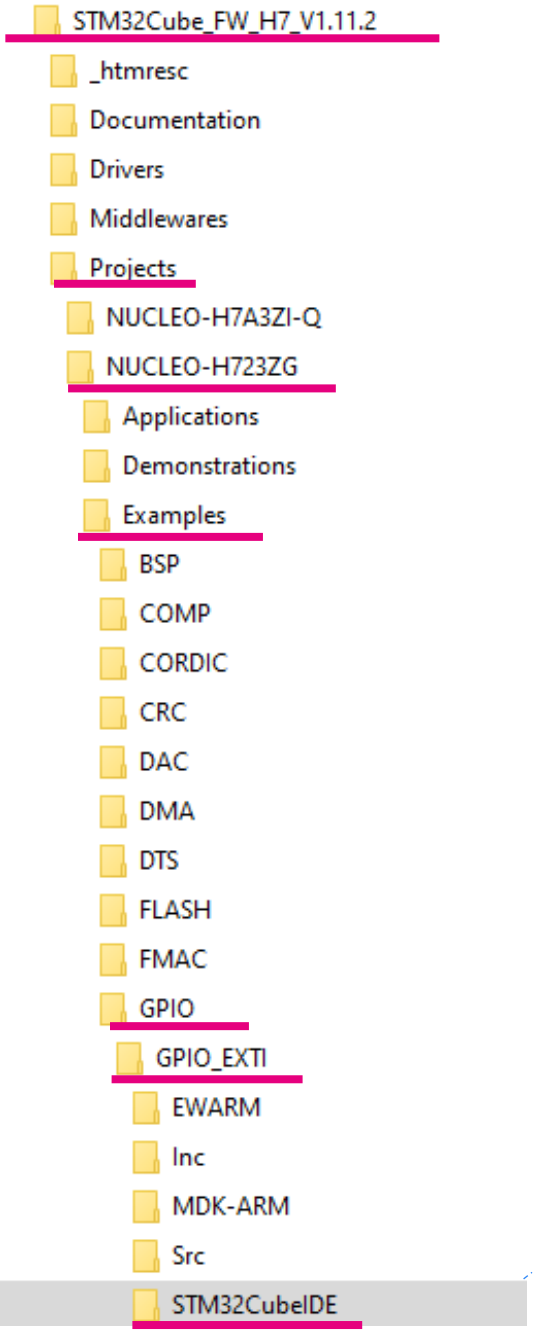




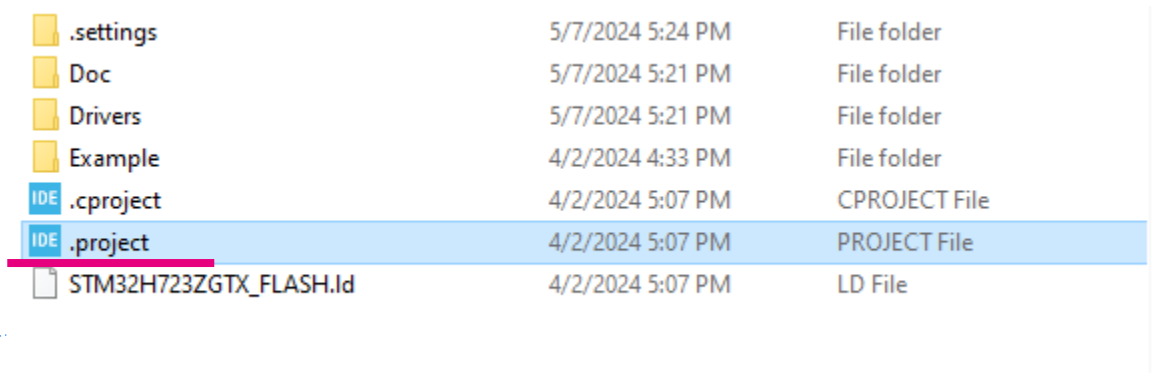
USBケーブルを使用してCN1コネクタとPCを接続するとボードを動作、デバックすることができます。

プログラム起動～ロード、デバックまで
順を追って説明します。

CubeIDE起動
サンプル･プログラムを展開



- 既存プロジェクトを開く
- ファイル＞開く＞ワークスペース にて以下のファイルを指定
  - STM32Cube_FW_H7_VX.X.X \Projects \NUCLEO-H723ZG\Examples \GPIO \GPIO_EXTI \STM32CubeIDE \.project

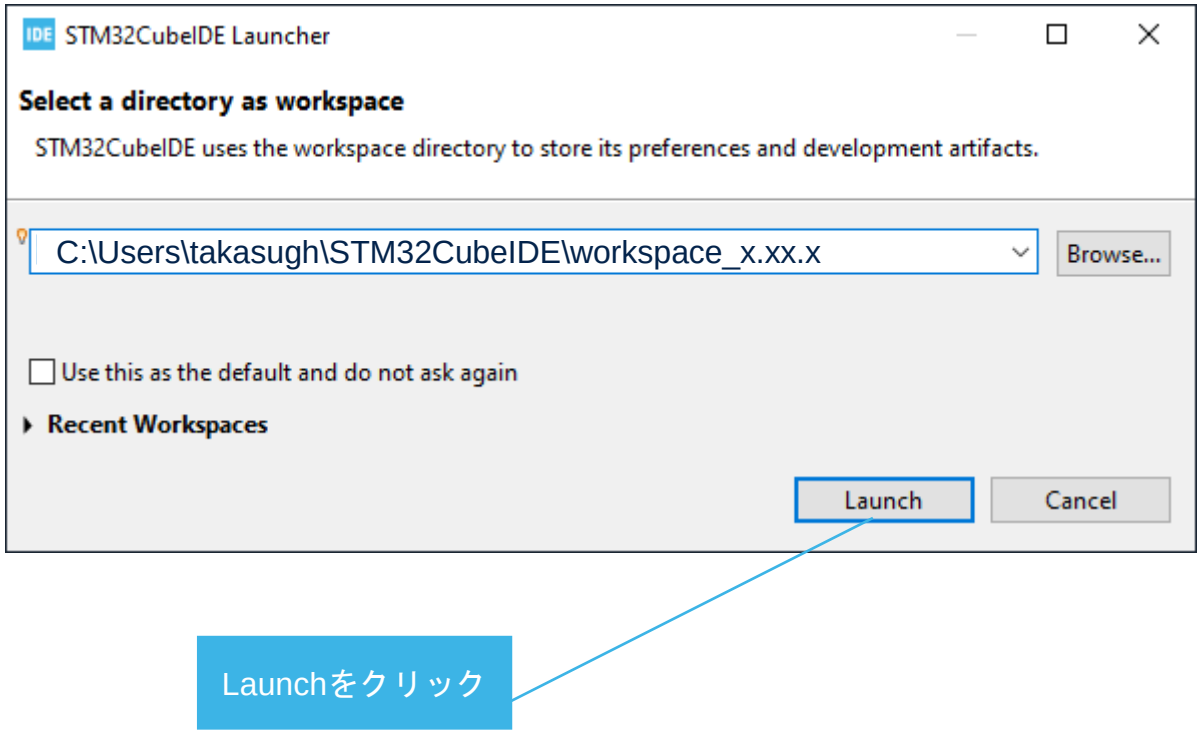


life.augmented

プログラム起動～ロード、デバックまで
順を追って説明します。

CubeIDE起動

- ワークスペースの確認画面



プログラム起動～ロード、デバックまで
順を追って説明します。

STM32CubeIDE 説明

STM32CubeIDE 起動画面

現在のワークスペースの名前

メニュー・バー

ツール・バー

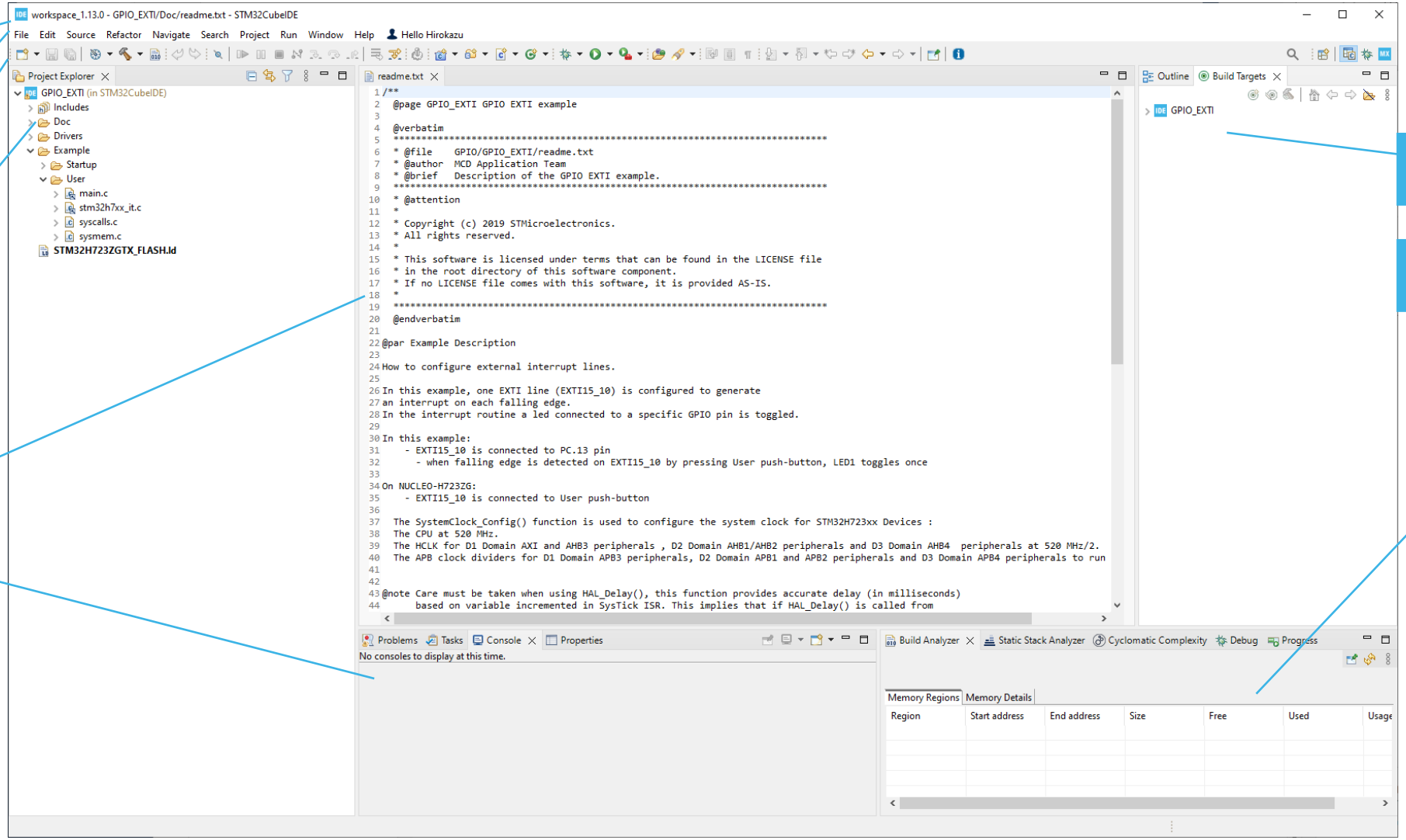
プロジェクト・エクスプローラ

エディター・エリア

コンソール・ビュー

アウトライン・ビュー

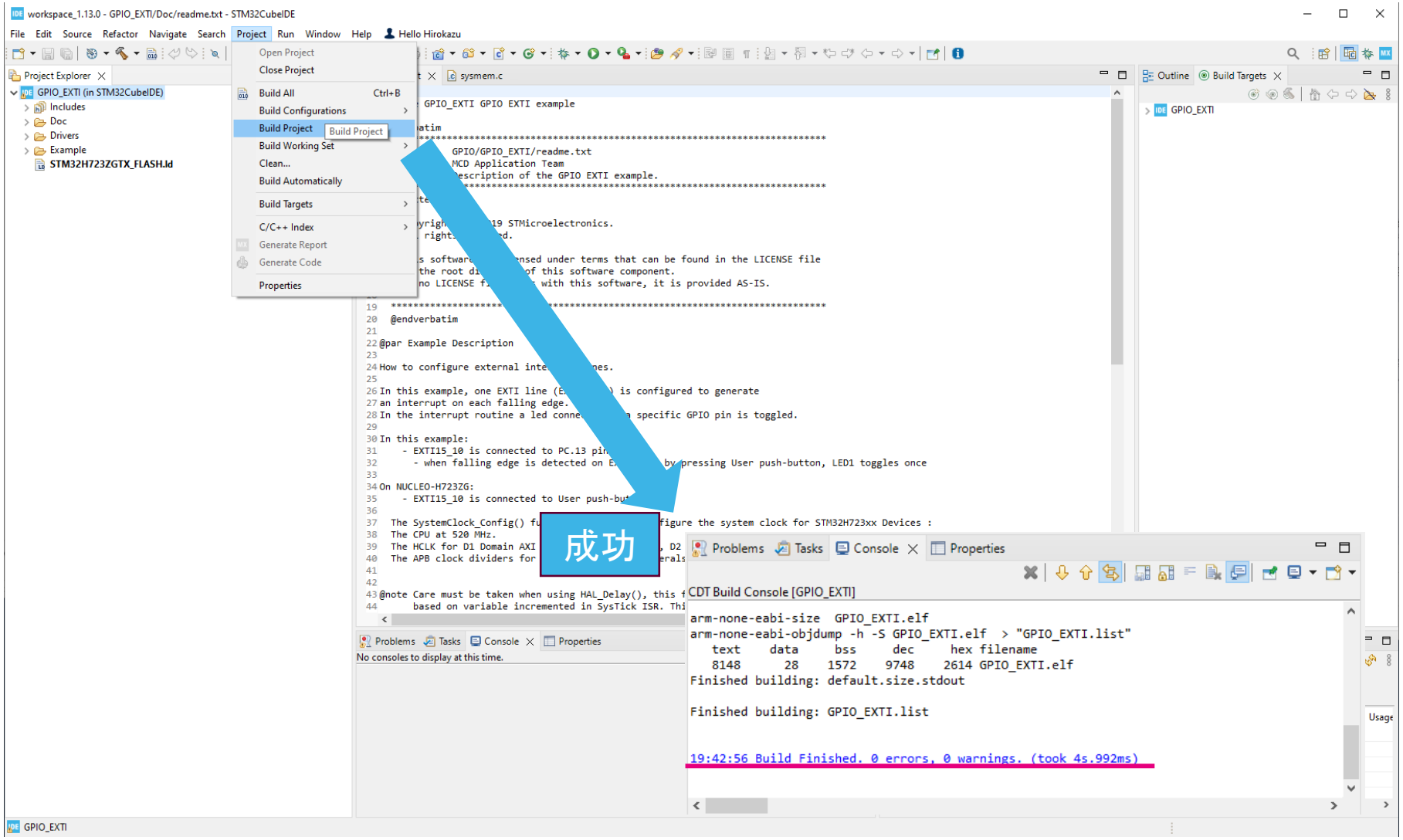
ビルド・アナライザ



プログラム起動～ロード、デバックまで
順を追って説明します。

ビルド（コンパイル）

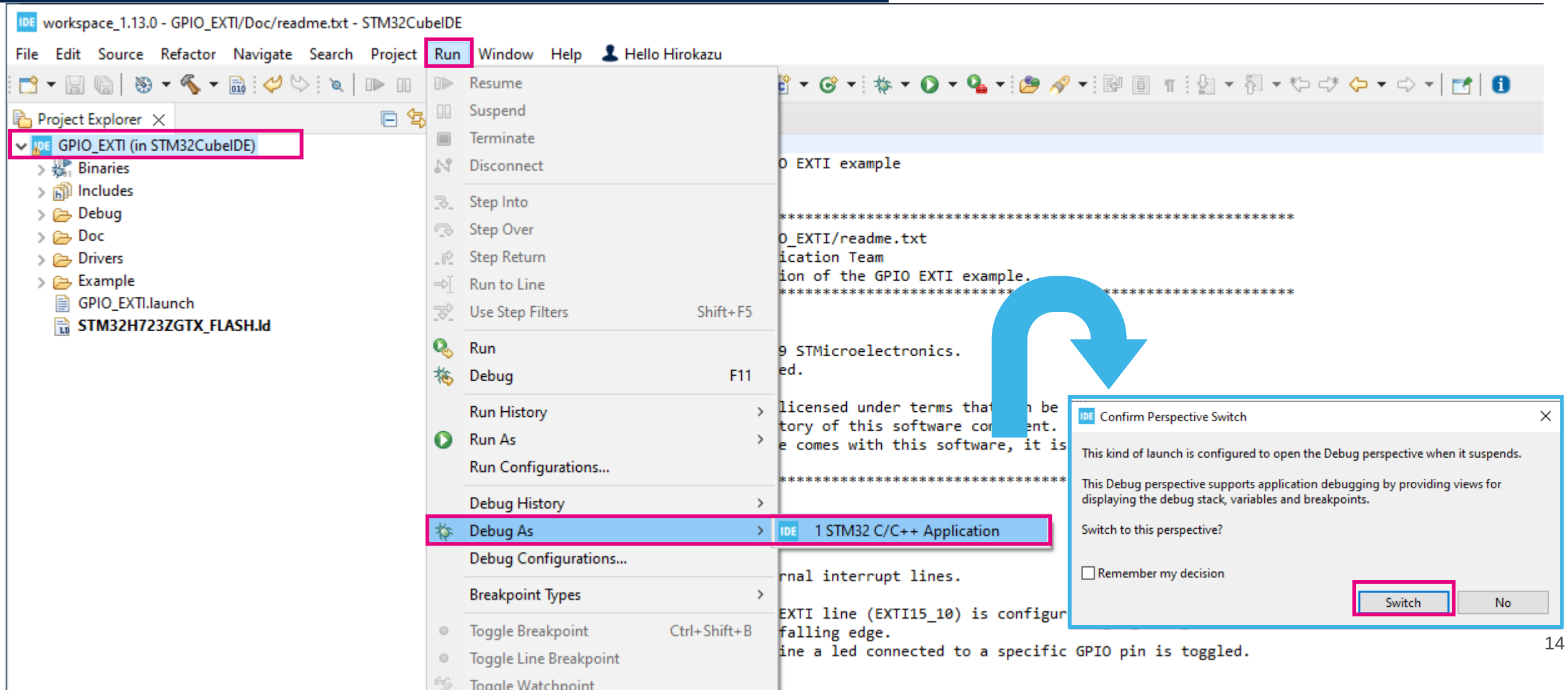
プロジェクトの再ビルドを行う



プログラム起動～ロード、デバックまで
順を追って説明します。

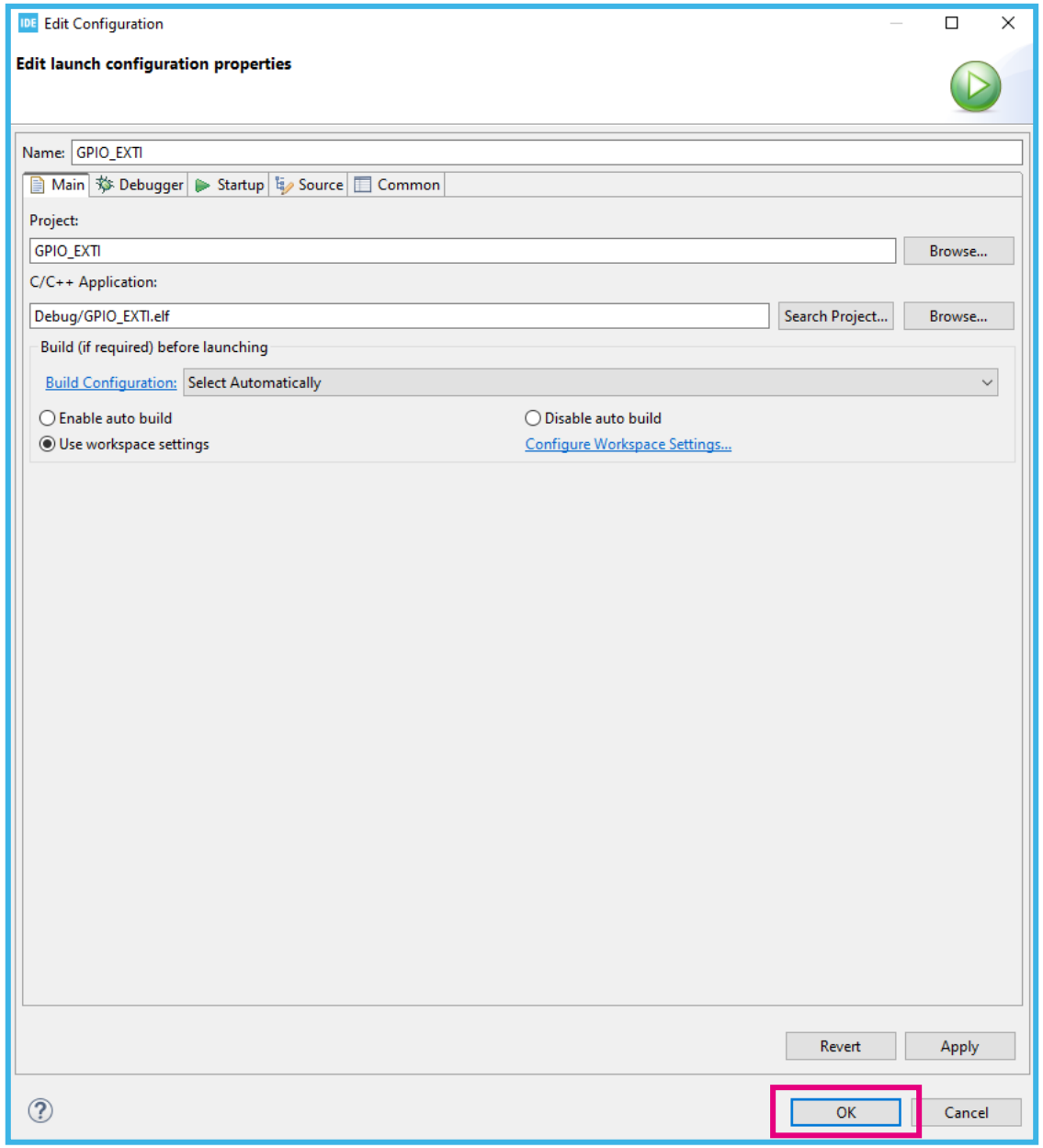
ロード

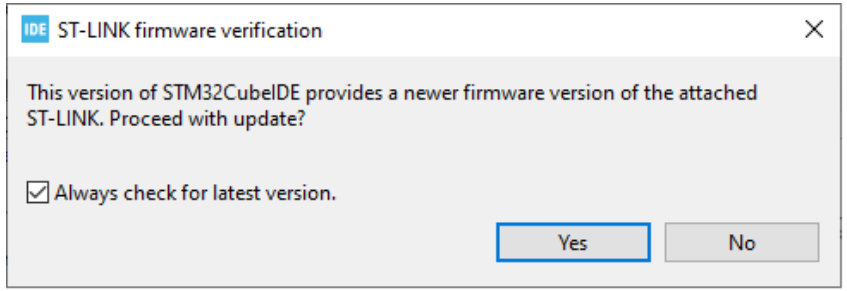
ターゲットにダウンロードしてデバッガを起動



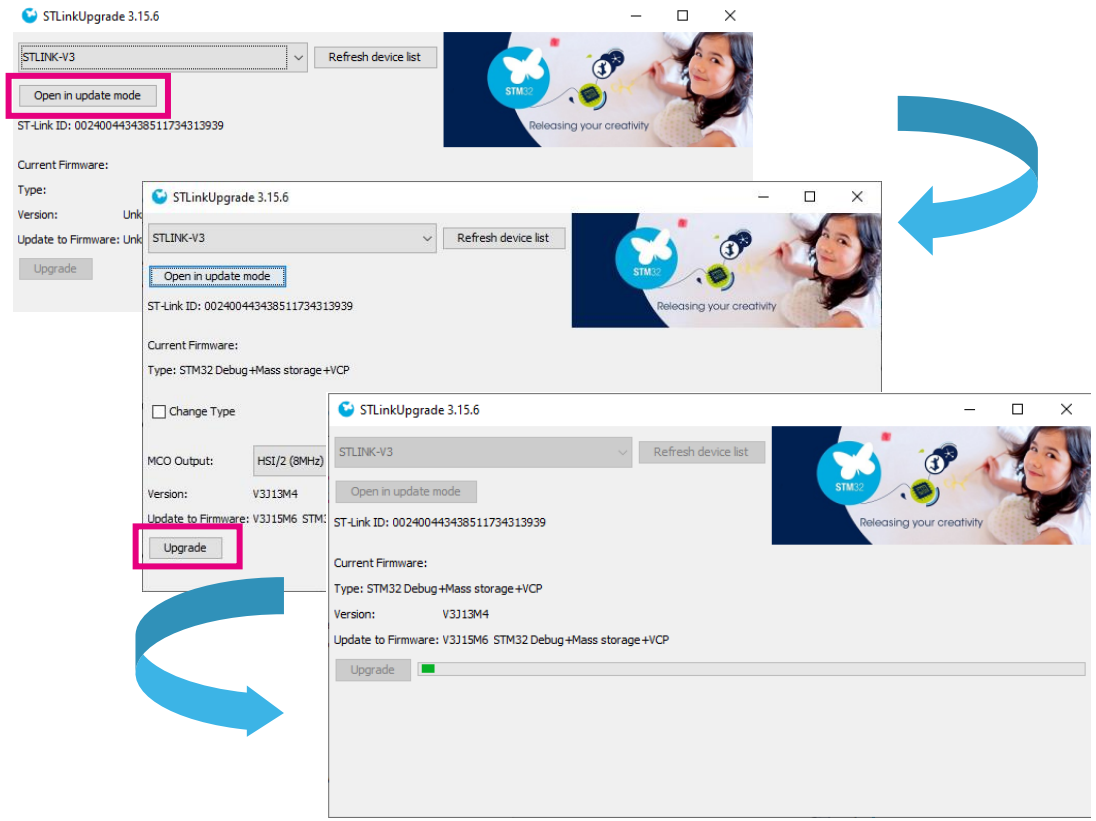
プログラム起動～ロード、デバックまで
順を追って説明します。

ロード



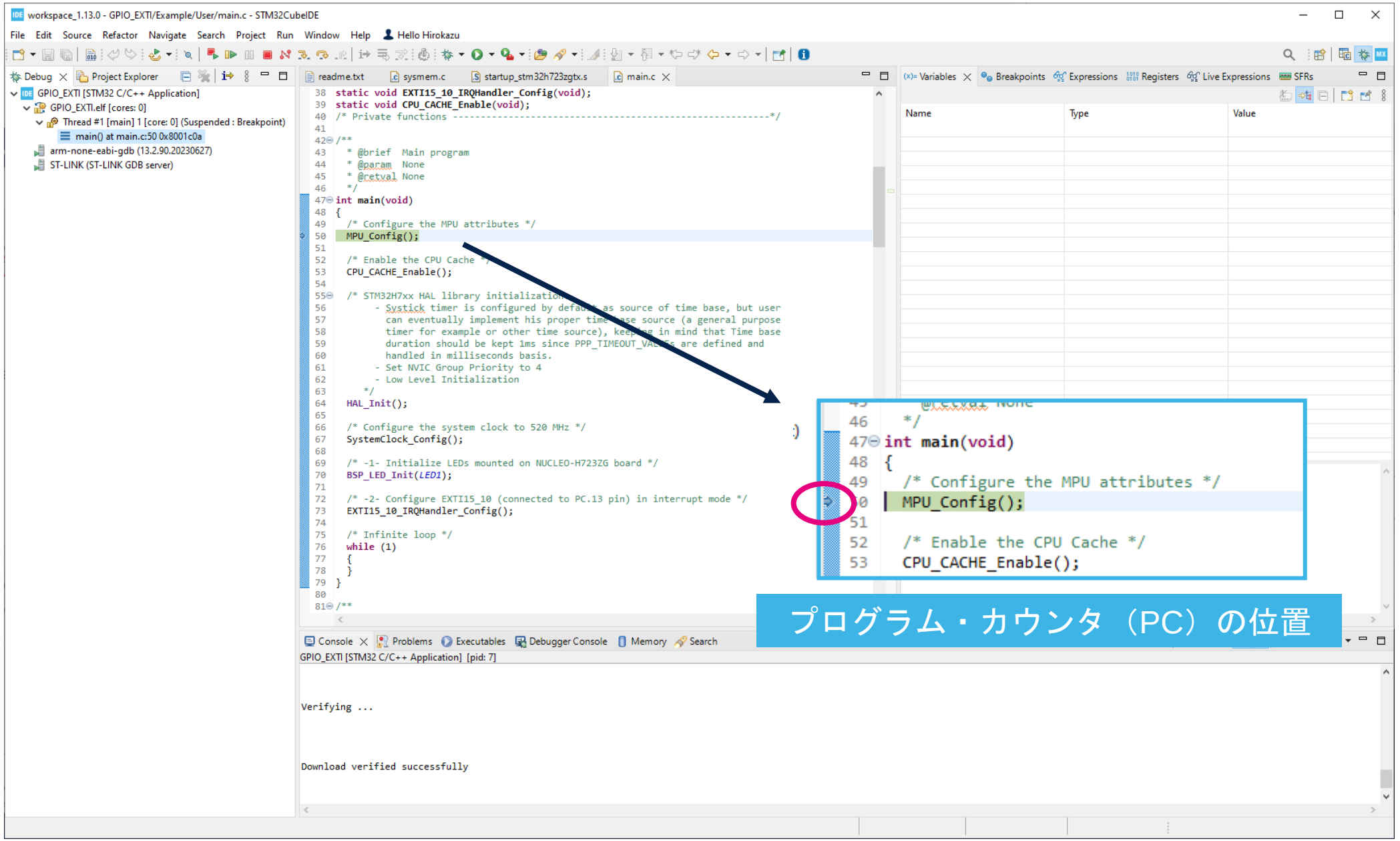


↑ST-Linkのファームウェアが古い場合こちらのメッセージが表示されます。YESをクリックして更新してください。



プログラム起動～ロード、デバックまで
順を追って説明します。

ロード&RUN



プログラム起動～ロード、デバックまで順を追って説明します。

ツール・バー 説明
ロード＆RUN

デバッグ用のツールバー

リジュームをクリックすると、プログラムがRUNします。
ボード上の青色ボタン（ユーザーボタン）を押す度に緑色LED（LD1）が点灯、消灯を繰り返します。

life.augmented

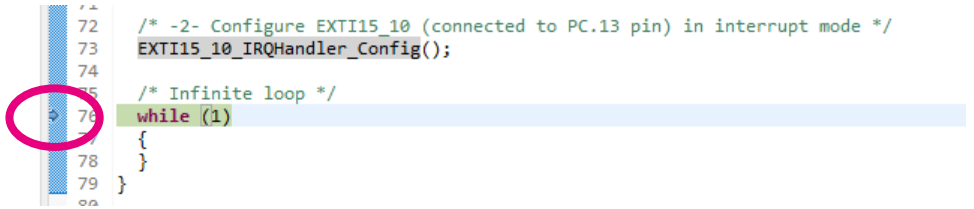
プログラム起動～ロード、デバックまで
順を追って説明します。

ツール・バー 説明
サスペンド

デバッグ用のツールバー

クリック

サスペンドをクリックすると、
プログラムが停止します。
停止すると、停止した位置の
プログラムが表示されます。



プログラム起動～ロード、デバックまで
順を追って説明します。

ツール・バー 説明
ステップ実行～Cソース

デバッグ用のツールバー

ステップイン、ステップオーバーでプログラムの実行内容をより深く見ていくことができます。

```
64   HAL_Init();
65
66   /* Configure the system clock to 520 MHz */
67   SystemClock_Config();
68
```

```
103 static void SystemClock_Config(void)
104 {
105   RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
106   RCC_OscInitTypeDef RCC_OscInitStruct = {0};
107   HAL_StatusTypeDef ret = HAL_OK;
108
109   /*!< Supply configuration update enable */
110   HAL_PWREx_ConfigSupply(PWR_LDO_SUPPLY);
```

ステップ・イン

```
67   SystemClock_Config();
68
69   /* -1- Initialize LEDs mounted on NUCLEO-H72
70   BSP_LED_Init(LED1);
71
```

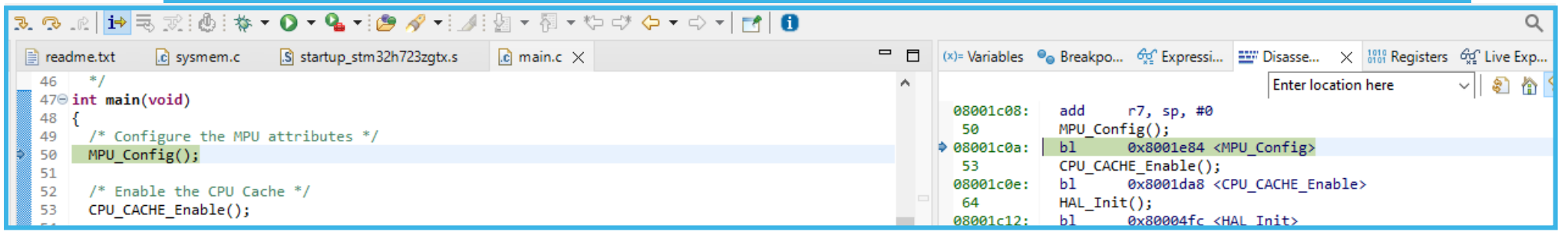
ステップ・オーバー

プログラム起動～ロード、デバックまで
順を追って説明します。

ツール・バー 説明
ステップ実行～アセンブラ

デバッグ用のツールバー

アセンブル言語レベルでのより細かい実行を確認することができます。



life.augmented

プログラム起動～ロード、デバックまで
順を追って説明します。

ツール・バー 説明
再スタート

デバッグ用のツールバー

CPUをリセットし、プログラムを再起動します

プログラム起動～ロード、デバックまで
順を追って説明します。

ツール・バー 説明
プログラム終了時

デバッグ用のツールバー

デバッカの接続を解除します。

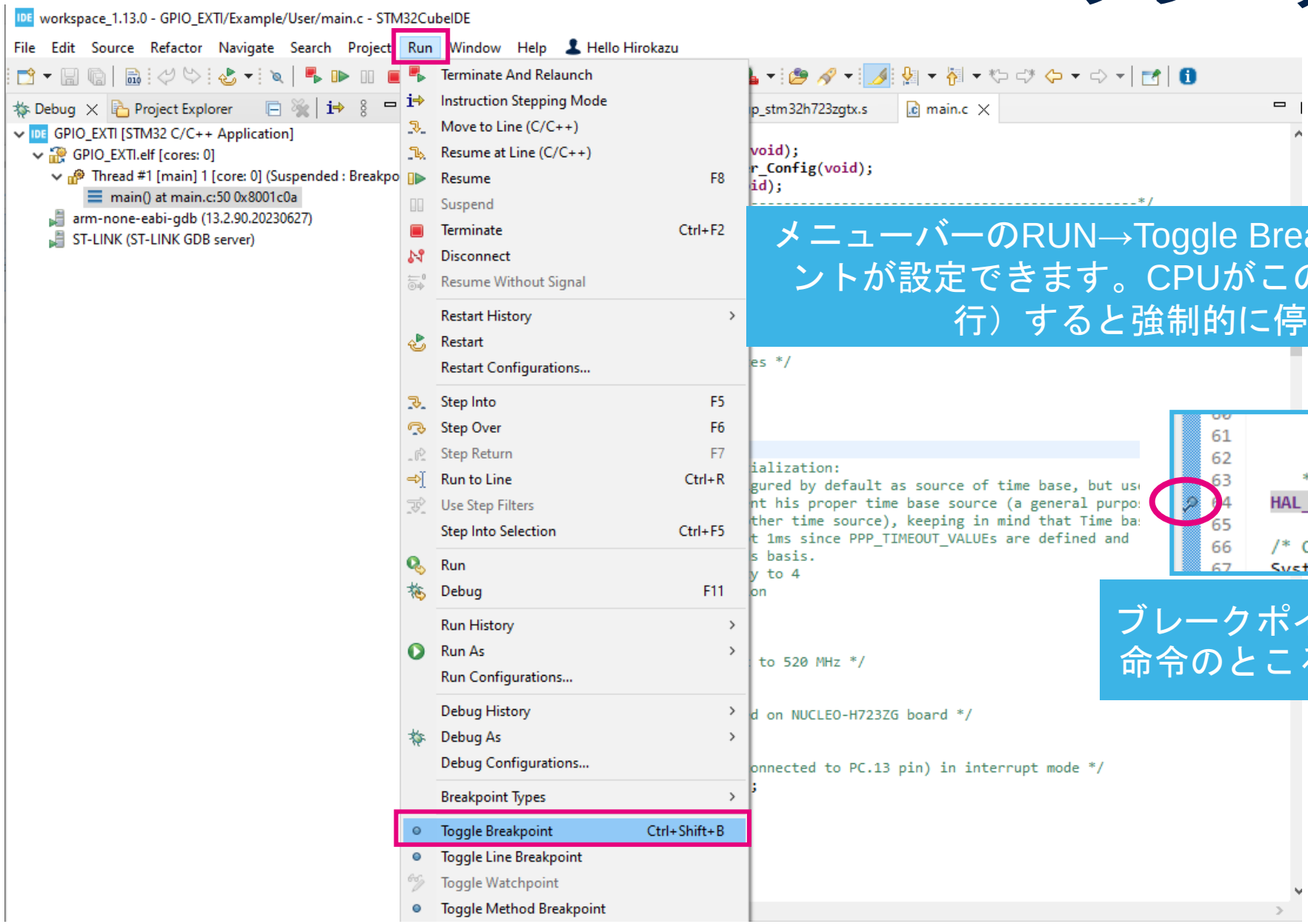
プログラム起動～ロード、デバックまで
順を追って説明します。

ツール・バー 説明
再接続

デバッグ用のツールバー

デバッカの接続を解除し再接続します。

ブレークポイント

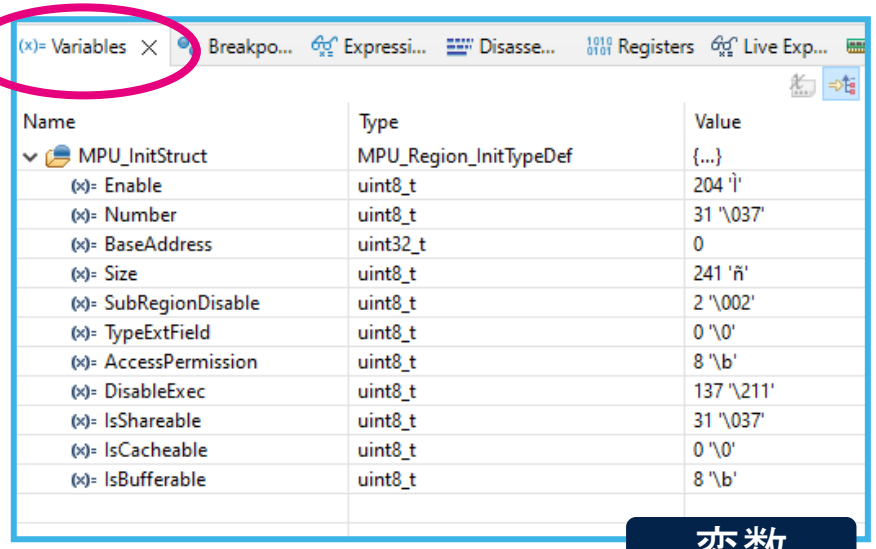


メニューバーのRUN→Toggle Breakpointでブレークポイントが設定できます。CPUがこのポイントを通過（実行）すると強制的に停止します。

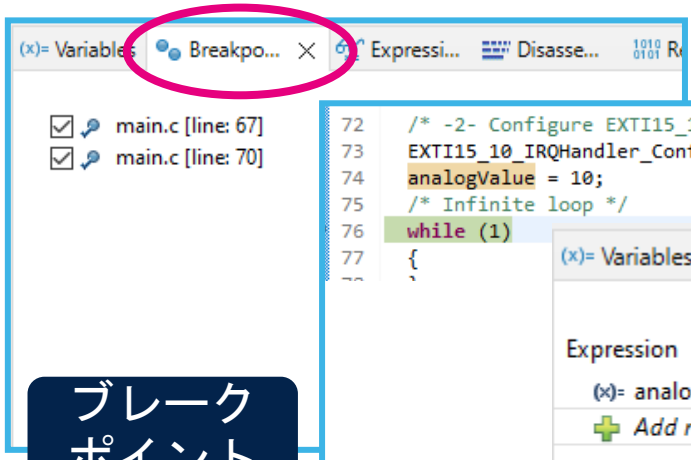
ブレークポイントが設定されると、命令のところに●が表示されます。

デバック

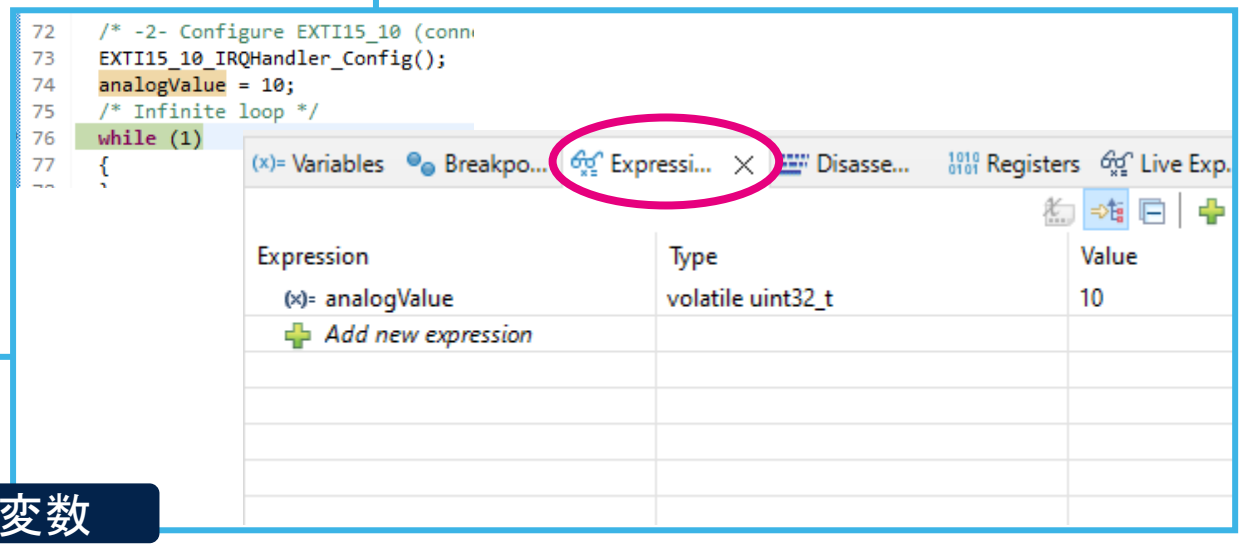
デバック中はアウトライン・ビューで確認可能



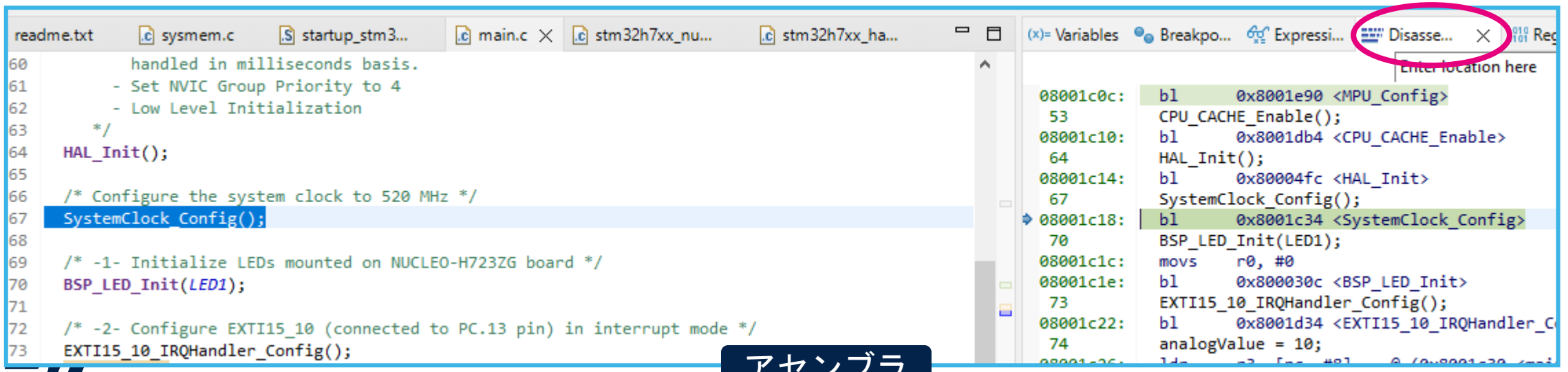
変数



ブレークポイント



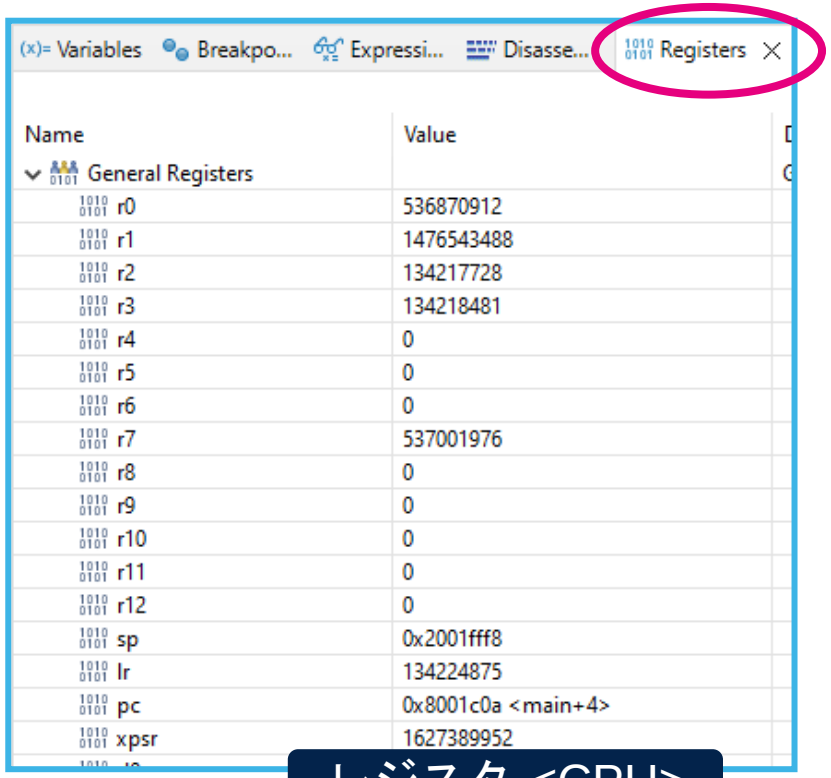
変数



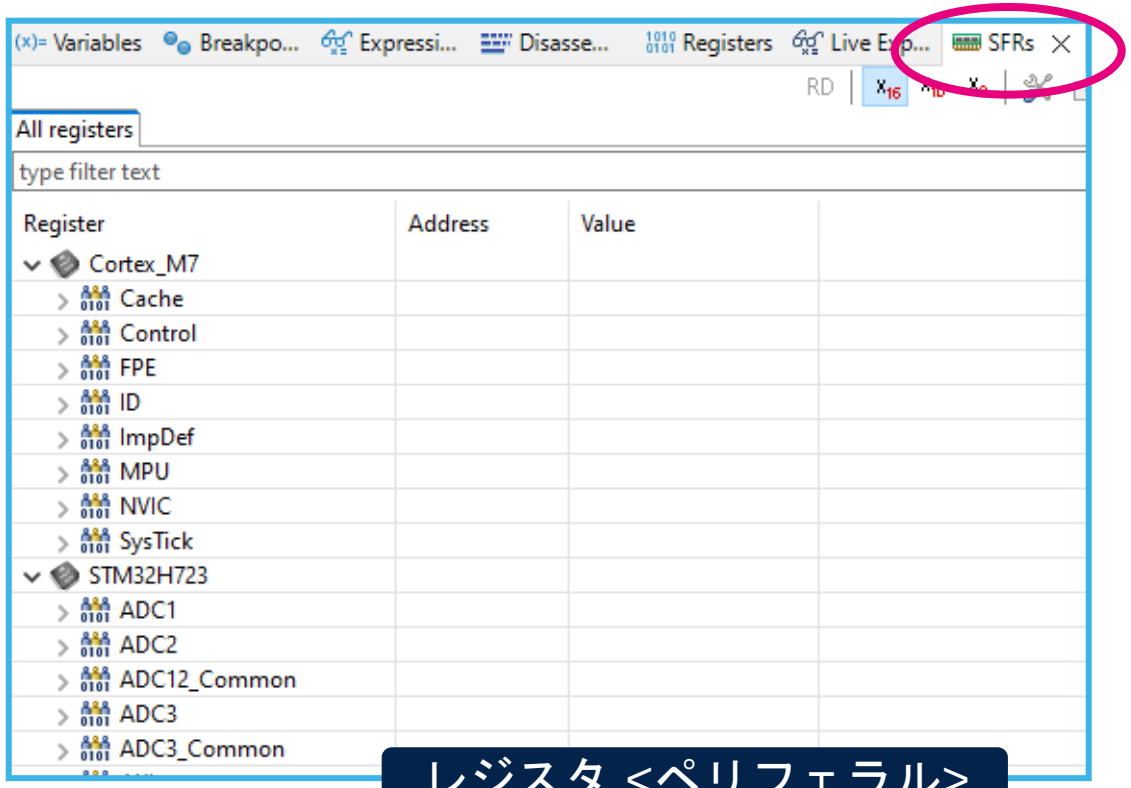
アセンブラ

デバック

デバック中はアウトライン・ビューで確認可能



レジスタ <CPU>



レジスタ <ペリフェラル>

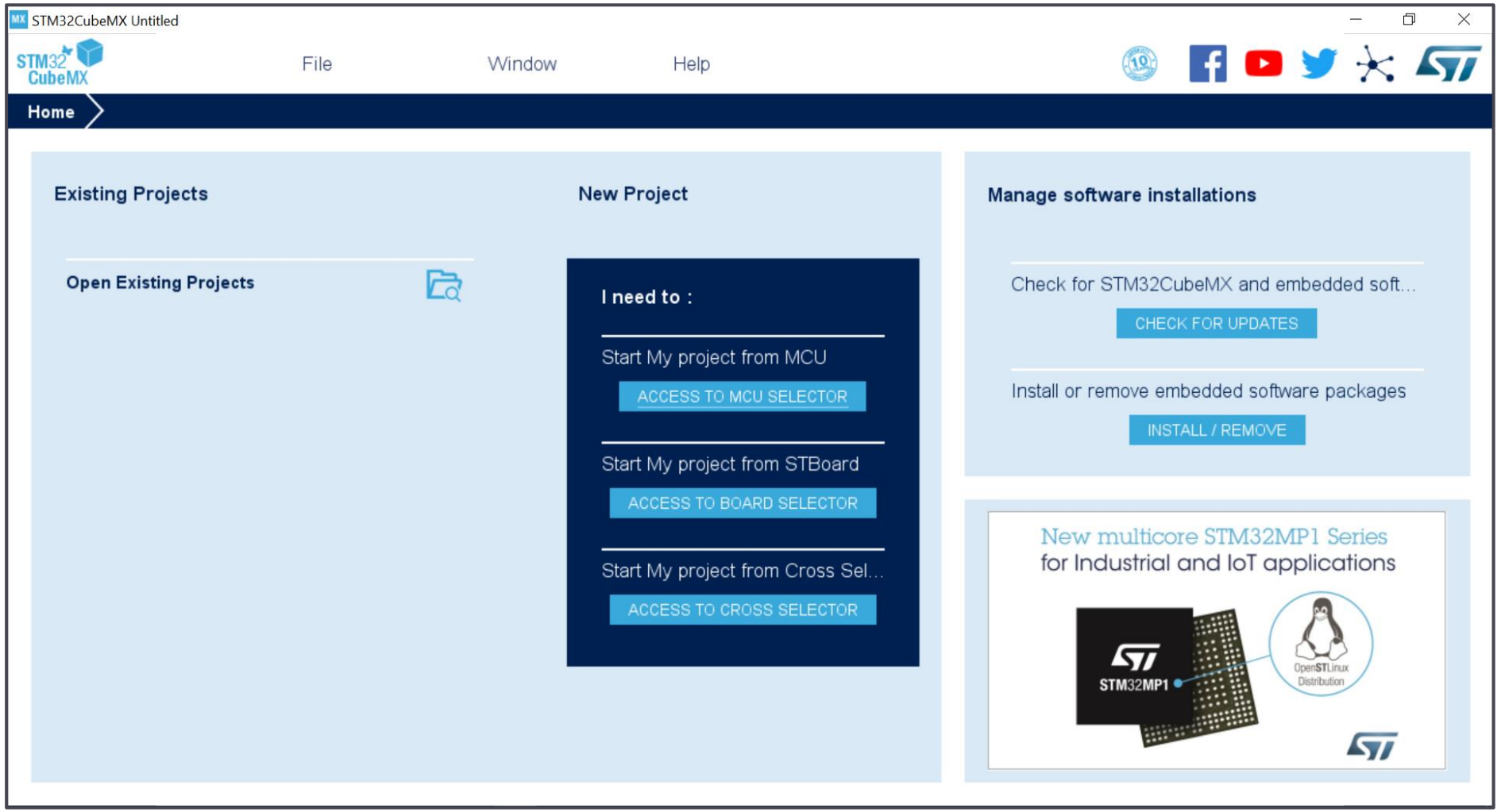
CubeMXとの連携

CubeMXを使用したプロジェクト作成例

セットアップ

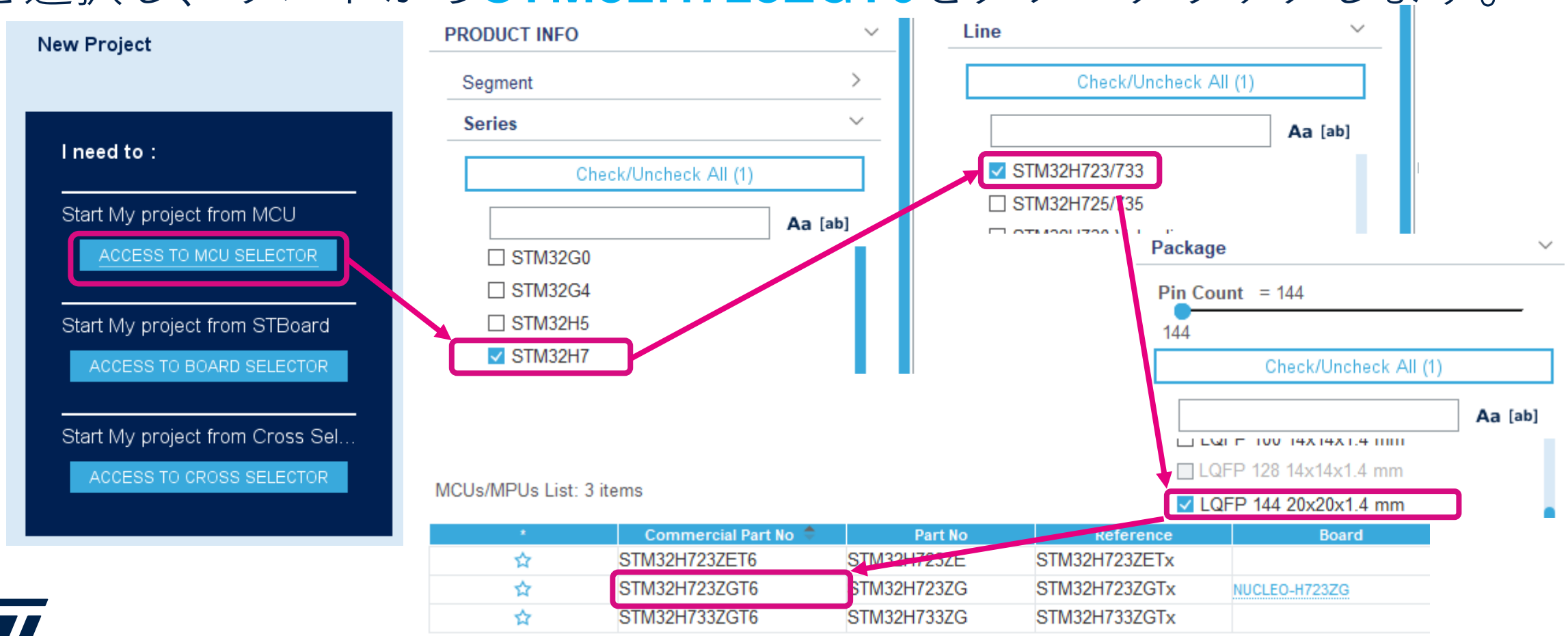
- STM32CubeMXを使用したGPIOトグルソフトウェアの作成
- サンプル･プログラム：汎用IOをトグルしてLEDを点滅するプログラム
- 手順
  ① NUCLEO-H723ZGを準備
  ② ボードとPCの接続
  ③ CubeMXでピンの設定
  ④ クロックの設定
  ⑤ プロジェクトの登録
  ⑥ コード生成

CubeMXの起動


STM32CubeMX Untitled
STM32 CubeMX
File
Window
Help
Home
Existing Projects
Open Existing Projects
New Project
I need to :
Start My project from MCU
ACCESS TO MCU SELECTOR
Start My project from STBoard
ACCESS TO BOARD SELECTOR
Start My project from Cross Sel...
ACCESS TO CROSS SELECTOR
Manage software installations
Check for STM32CubeMX and embedded soft...
CHECK FOR UPDATES
Install or remove embedded software packages
INSTALL / REMOVE
New multicore STM32MP1 Series
for Industrial and IoT applications
STM32MP1
OpenSTLinux Distribution

CubeMXでのデバイス選定

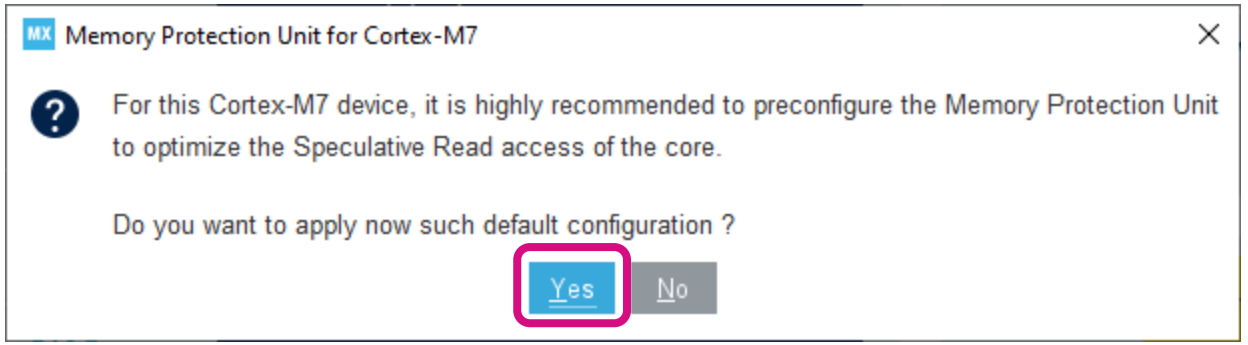
ACCESS TO MCU SELECTORから
Series：**STM32H7** Lines：**STM32H723/733** Package：**LQFP144**
を選択し、リストから**STM32H723ZGT6**をダブルクリックします。



life.augmented

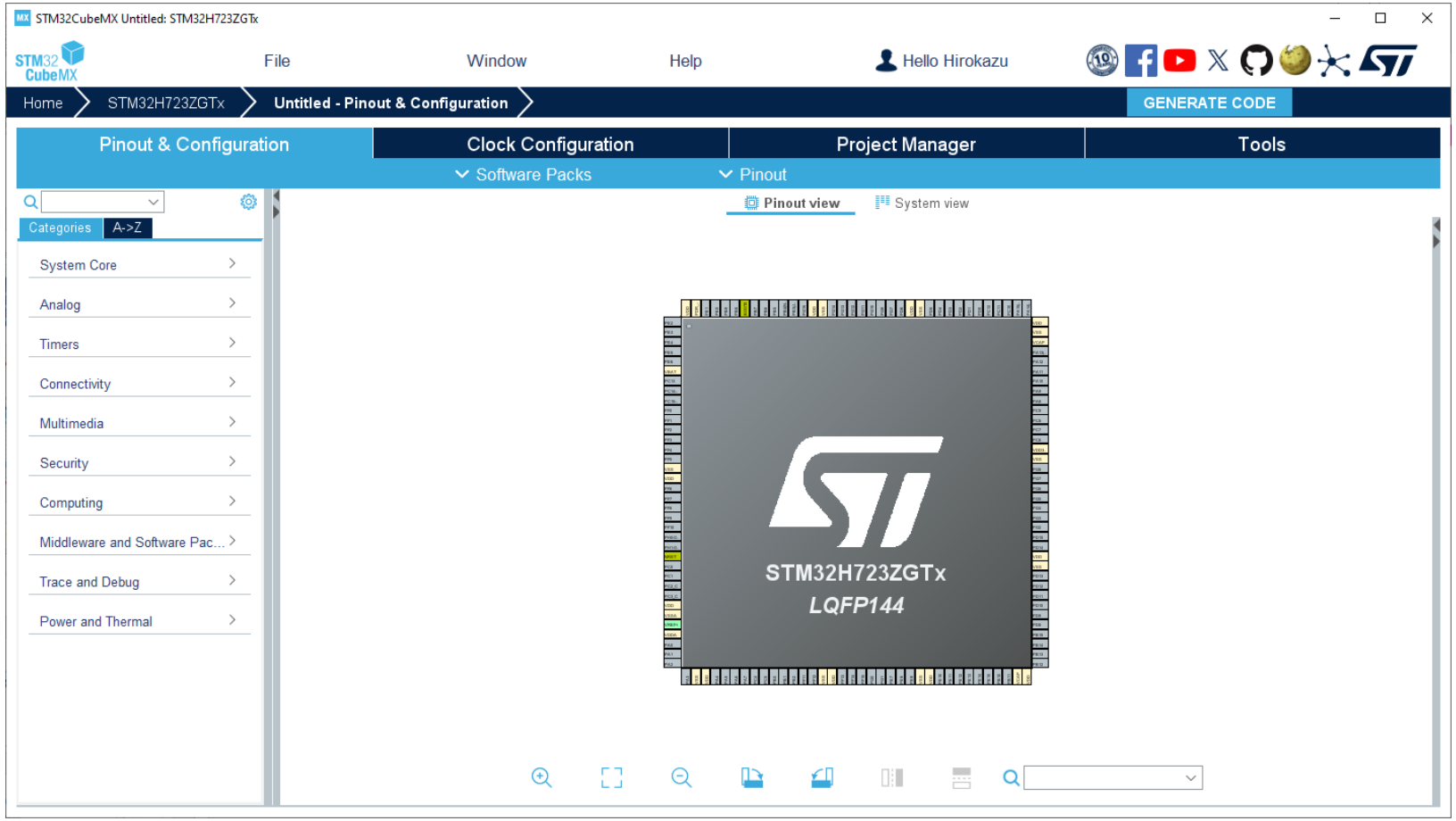
CubeMXでのデバイス選定

- このメッセージが出てくるのでYesをクリック



設定

プロジェクト画面が開きます。
ここから設定を進めていきます。



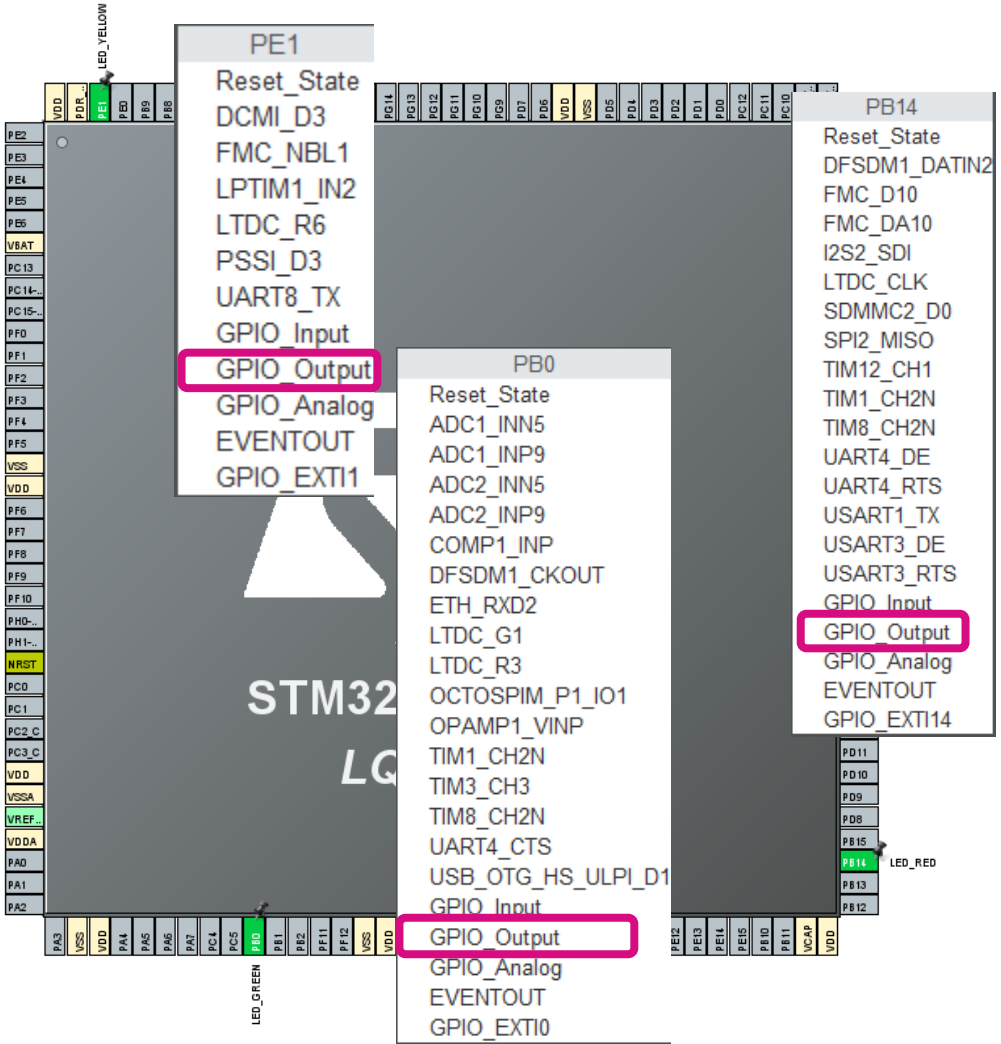
ピンの選択

以下のピン設定を行います。
設定はマイコンイメージのピン部分をクリックすることで
設定可能です。

PB0:LED_GREEN（LD1）
　　GPIO_Output

PE1:LED_YELLOW (LD2)
　　GPIO_Output

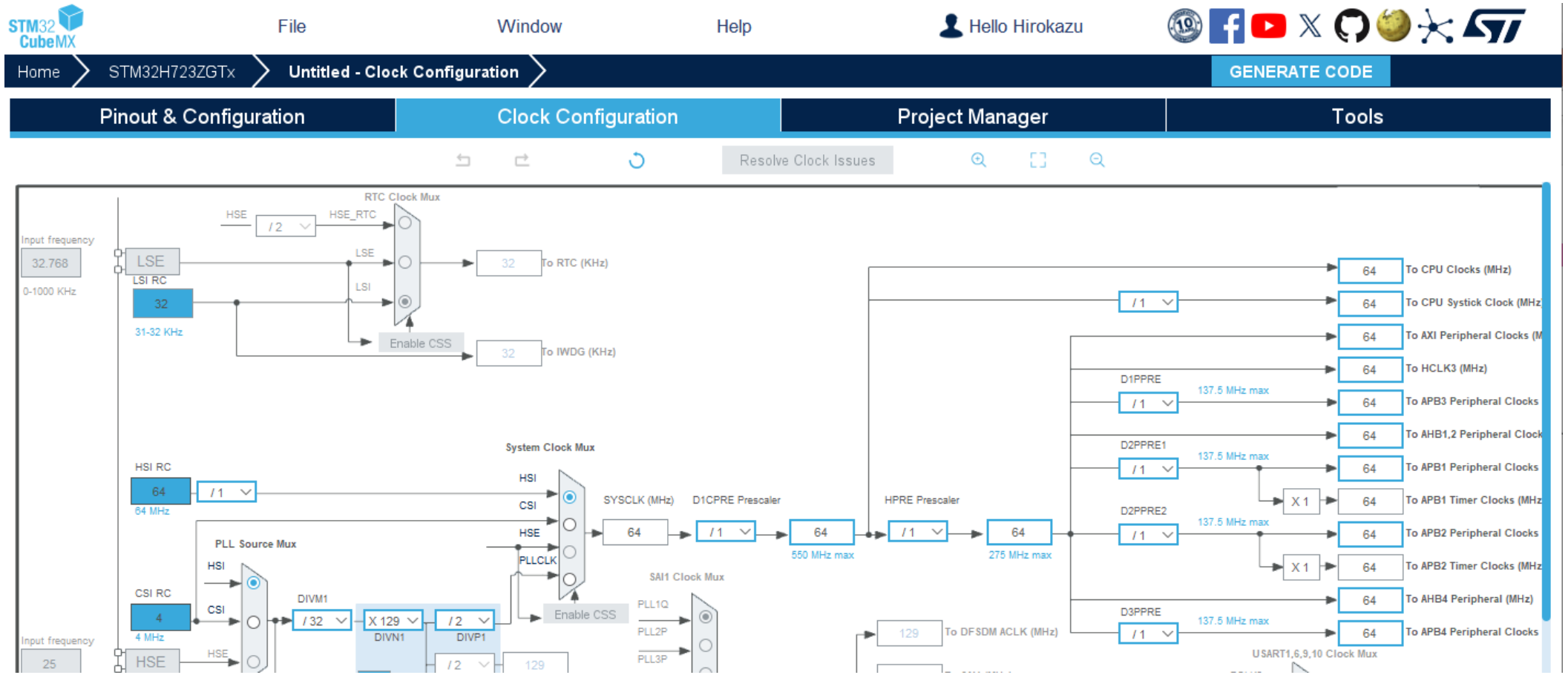
PB14:LED_RED (LD3)
　　GPIO_Output



設定されたピンは色が変わります。

クロックの設定

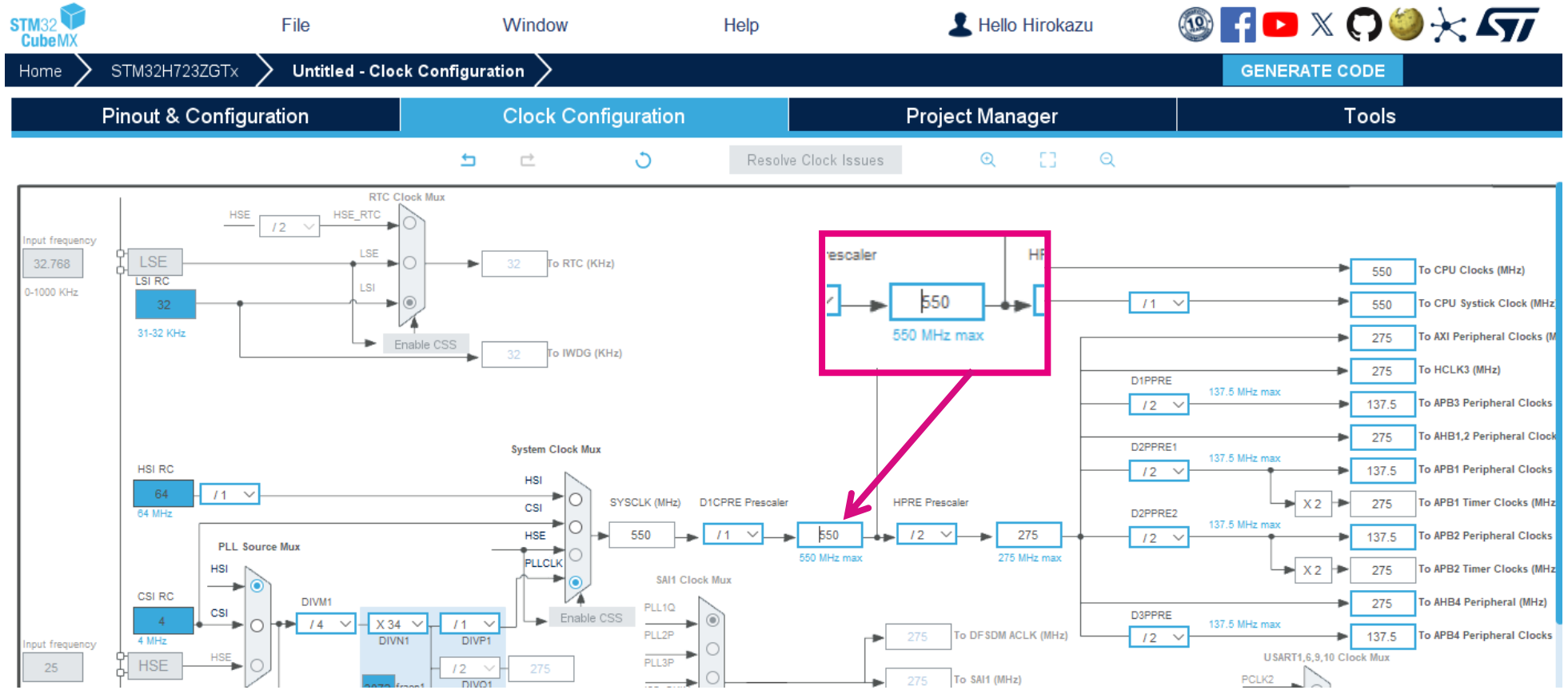
次にクロックの設定を行います。
Clock Configurationのタブを開きます。



クロックの設定

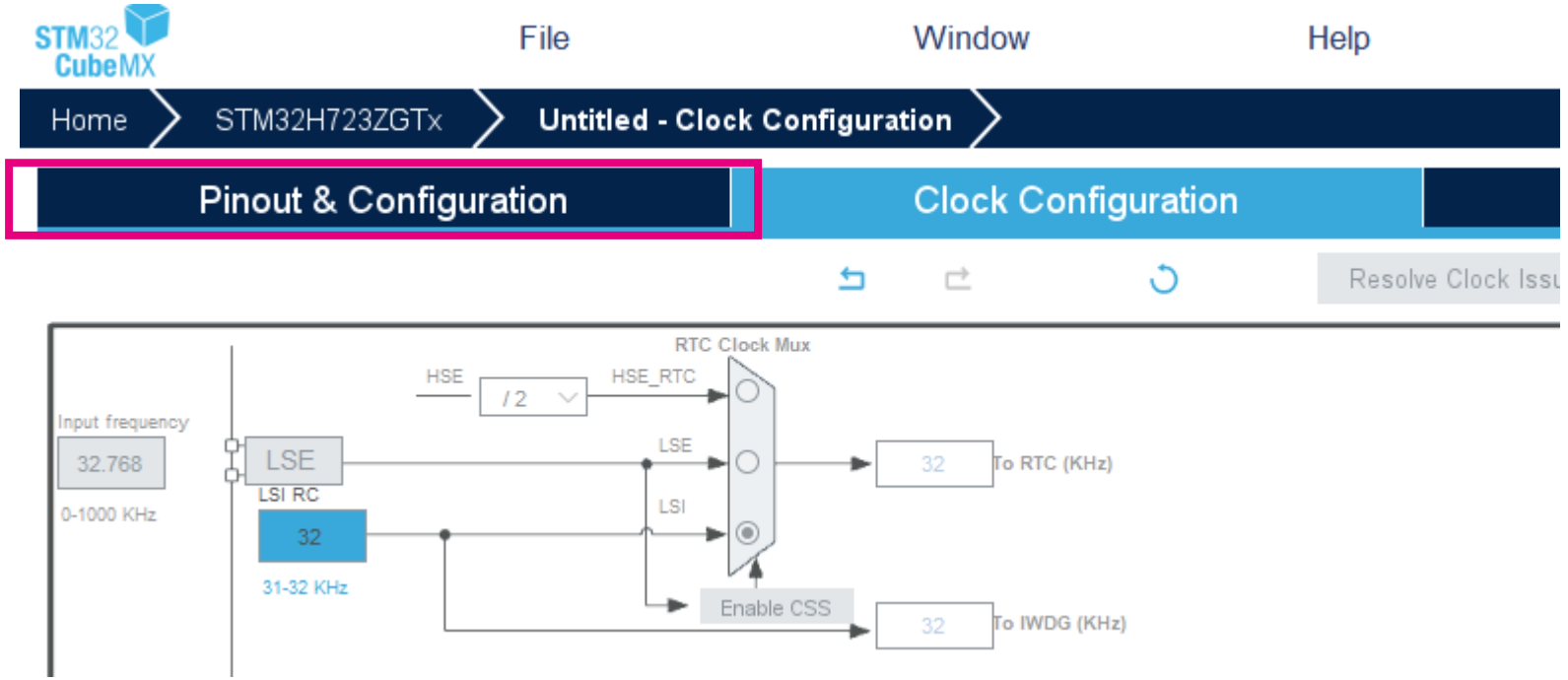
以下のようにクロックの設定を行います。

HCLK　64 → 550　本設定で、CPUクロック550MHz
ペリフェラルクロック275MHz/137.5MHz
に設定されます。



ピンの設定

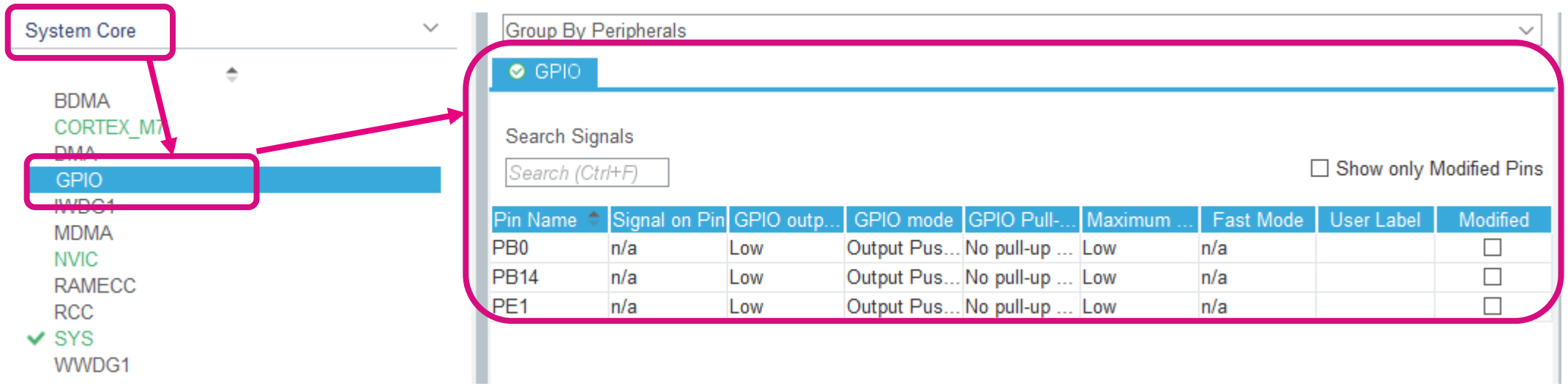
ピン設定の詳細を設定しますので、
再度**Pinout & Configuration**タブを開きます。



ピンの設定

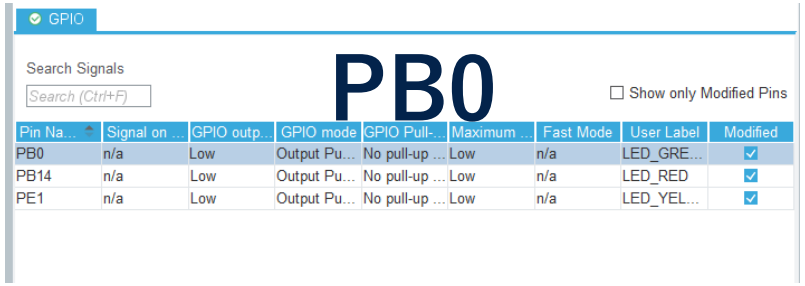
System Core→**GPIO**→GPIO Mode and Configuration

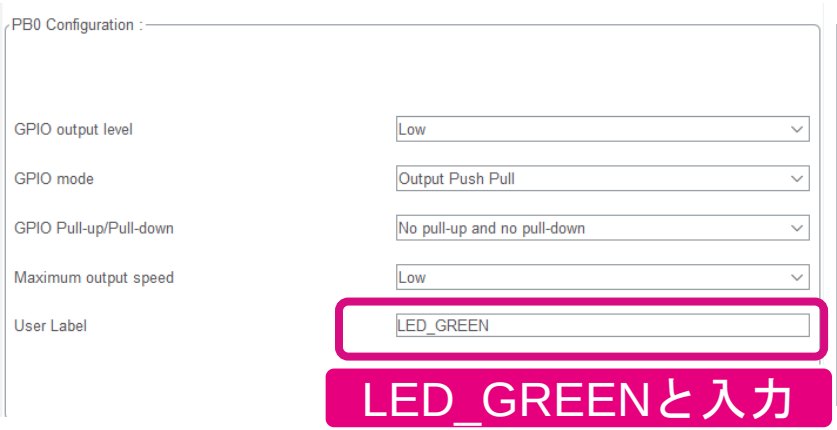
各Pinの詳細設定を行います。

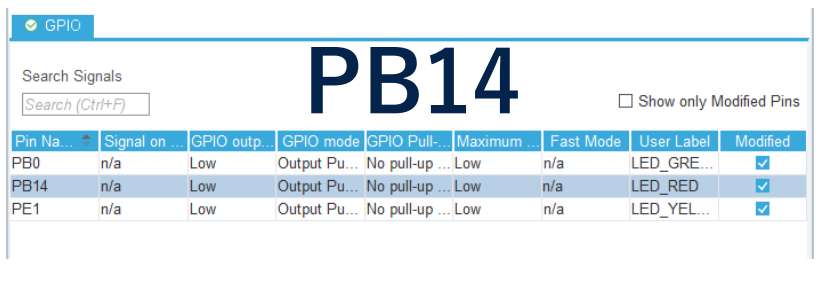


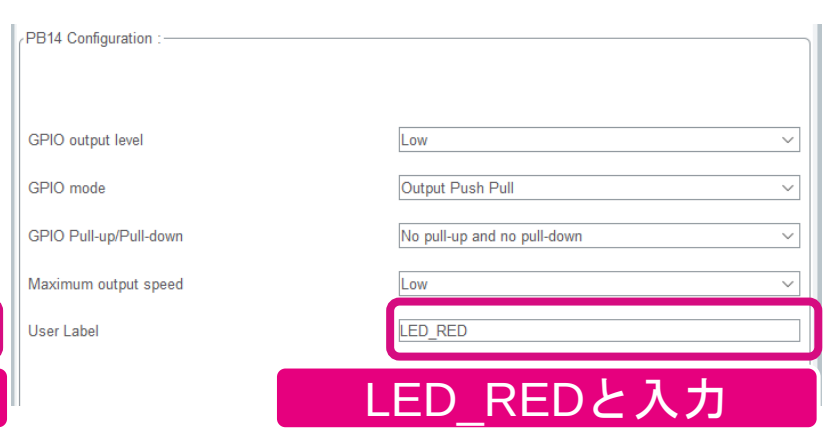
ピンの設定

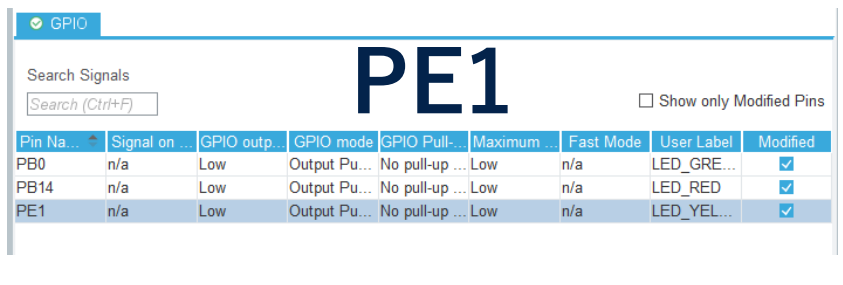
リストの各ピンを選択し詳細設定を行います。
PB0、PB14およびPE1の設定を以下のように設定します。

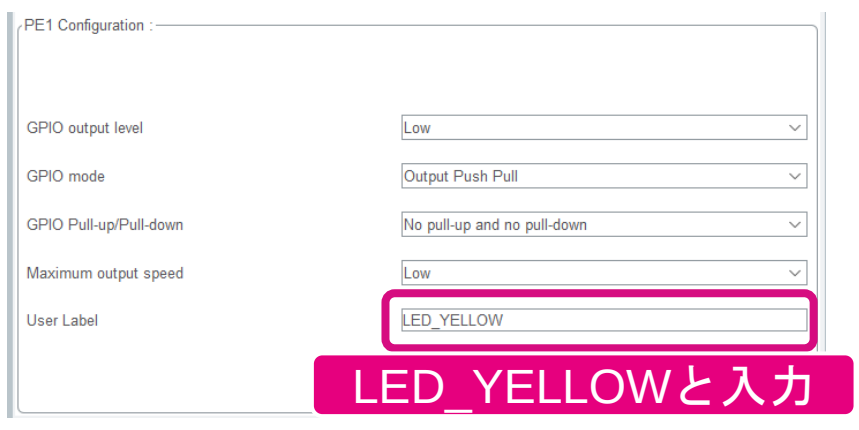












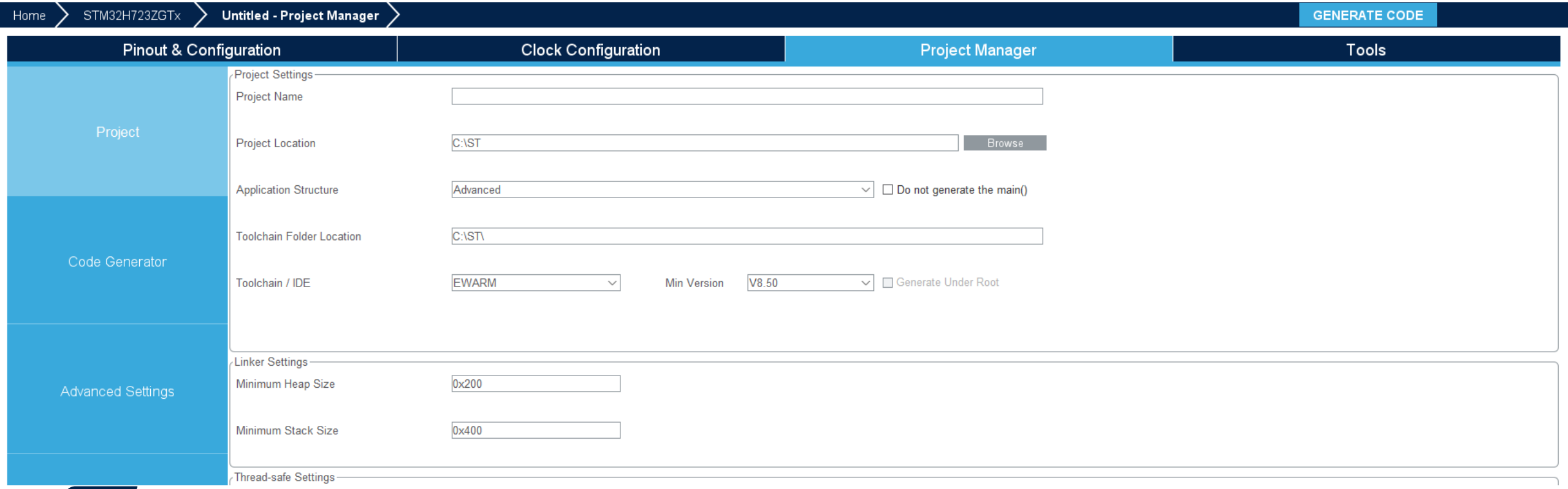
life.augmented

プロジェクト選択

ここまででマイコンの設定は完了ですので、次は

Project Manager

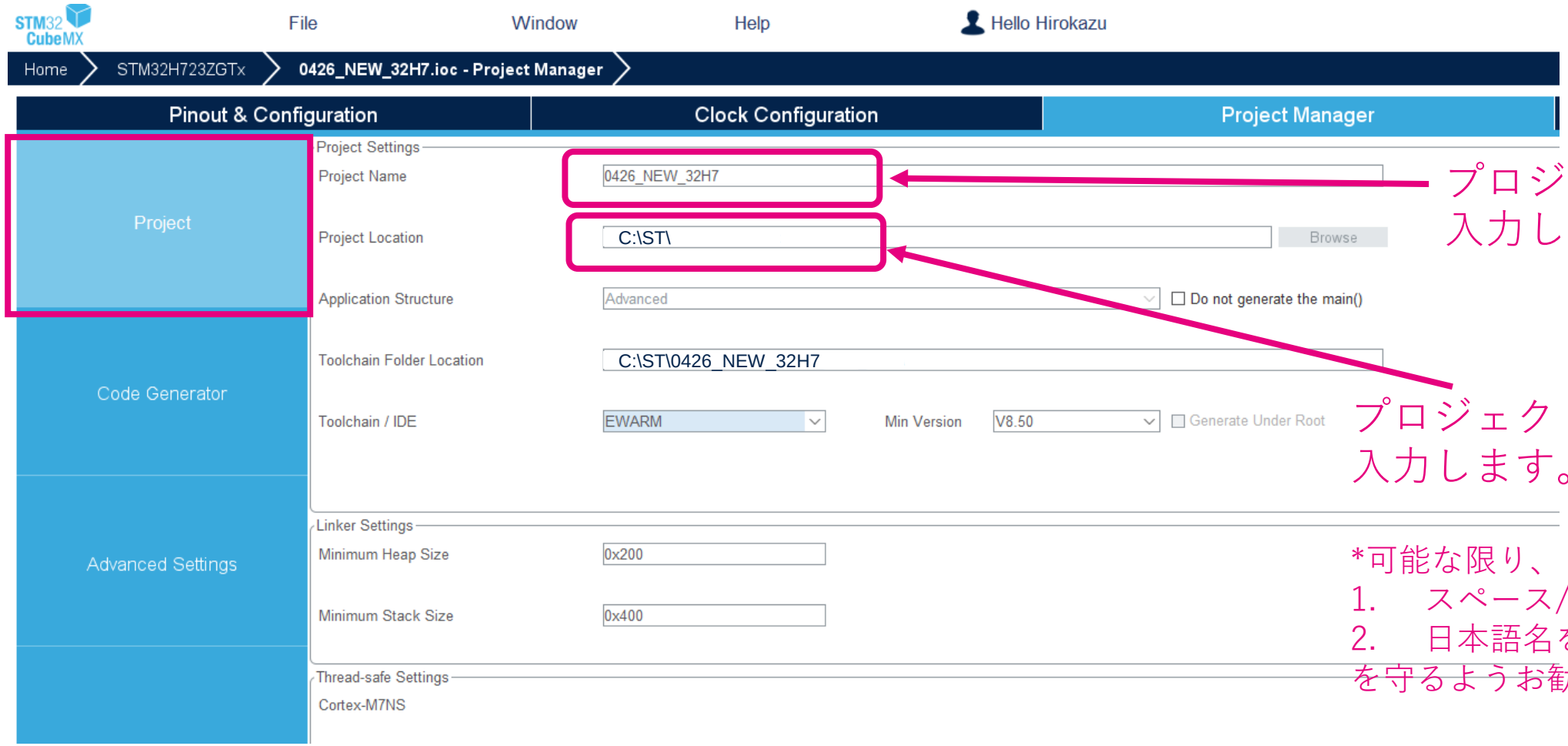
プロジェクトの名前や保存先等を設定します。



life.augmented

Project プロジェクト選択

プロジェクトの設定を行います。



プロジェクトの名称を
入力します。

プロジェクトの保存先を
入力します。

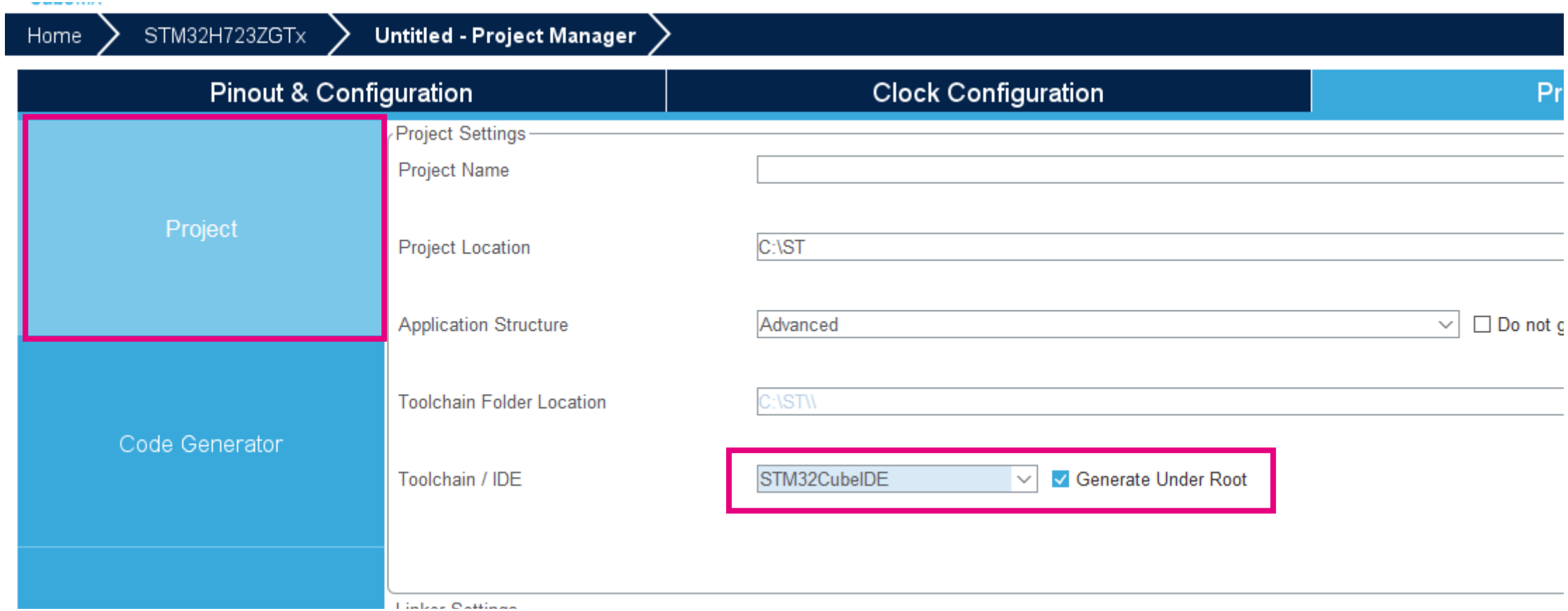
*可能な限り、フォルダ名に関して、
1. スペース/空白文字を含めない
2. 日本語名を含まない
を守るようお勧めします。

life.augmented

プロジェクト選択

Project

Toolchain / IDEを「STM32CubeIDE」に設定します。

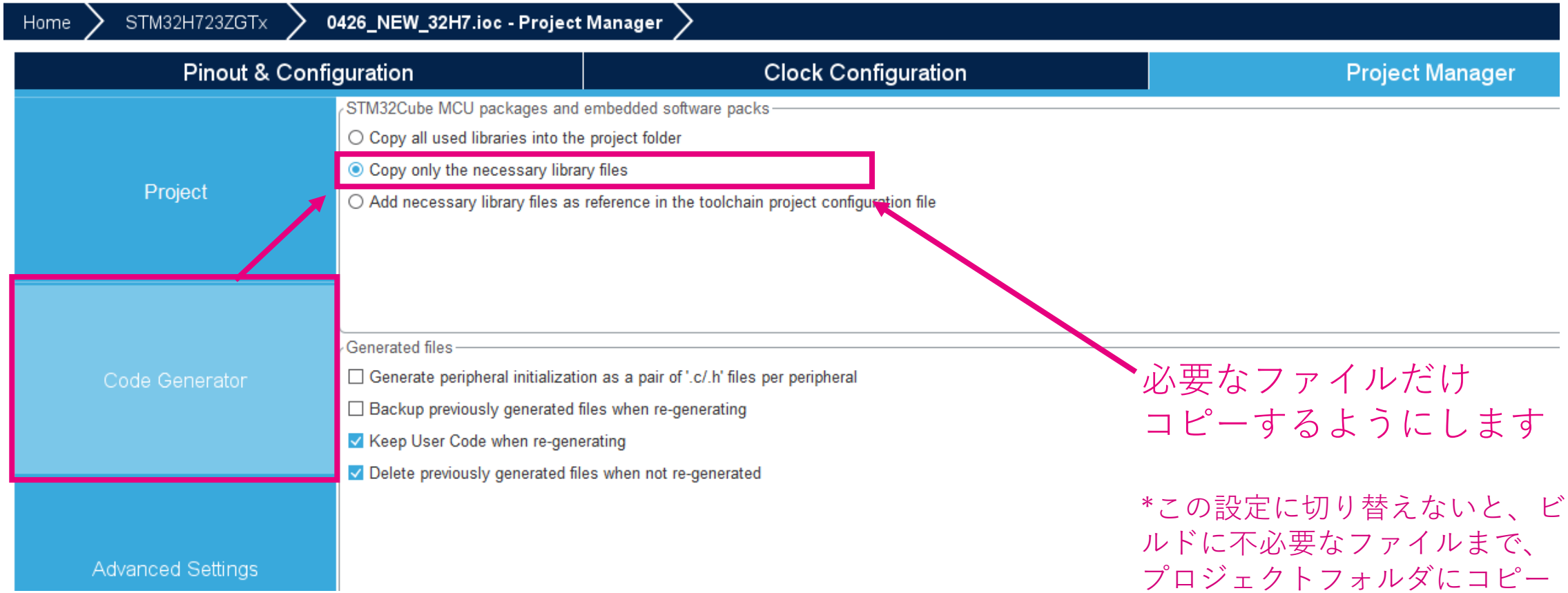


life.augmented

プロジェクト選択

Code Generator

コード生成に関する設定を行います。



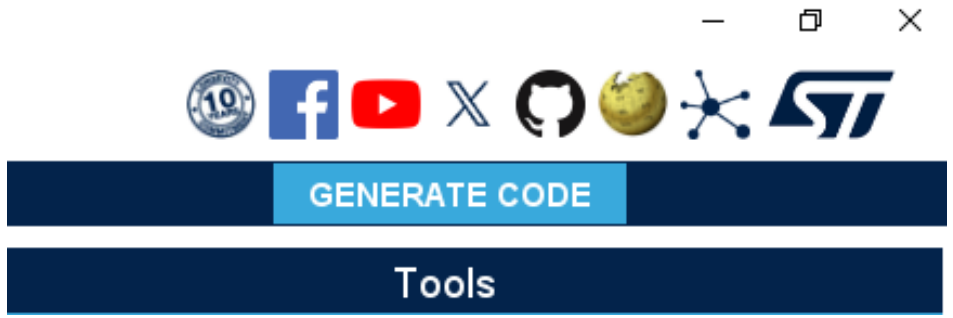
必要なファイルだけ
コピーするようにします

*この設定に切り替えないと、ビルドに不必要なファイルまで、プロジェクトフォルダにコピーされ、PCのストレージを圧迫します。

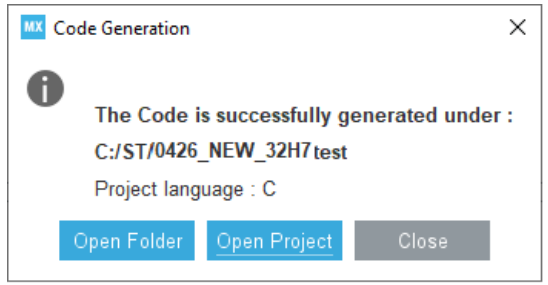
コード生成

お疲れさまでした！

ここまでに設定した内容で、コードを生成します。
GUIの右上の**GENERATE CODE**ボタンを押します。



コード生成が完了すると以下のダイアログが開きます。



Open Projectボタンを押下することでSTM32CubeIDEが開きます。

life.augmented

CubeIDE コード追記

ここからユーザ処理を追加していきます。
編集するファイルはmain.cになります。
STM32CubeMXで生成したコードには、
/* USER CODE BEGIN xxx */　と　**/* USER CODE END xxx */**
の記載があり、このコメントの範囲内に
ユーザコードを追加していきます。

上記のルールに則って追記したユーザーコードは、
再度CubeMXにてコード生成した場合でも保護されます。

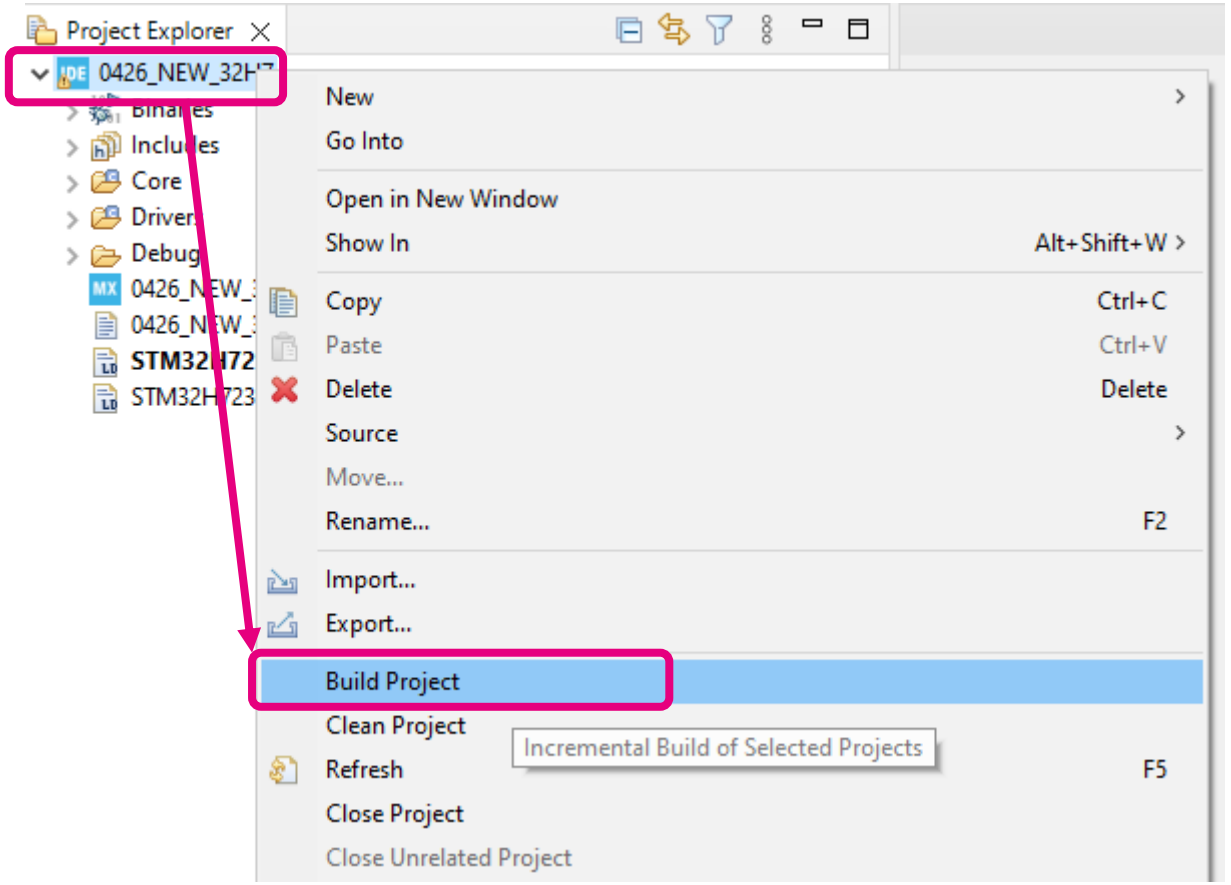
life.augmented

CubeIDE コード追記

STM32CubeIDEが起動したら、ビルドが通ることを確認します。

プロジェクトを右クリック → Build Project

右クリック



CubeIDE コード追記

追記箇所
/* USER CODE BEGIN **3** */ ～ /* USER CODE END **3** */
ループ処理を追加します。
HAL__Delay関数を使用して定期的にGPIOをトグルします。

```
/* USER CODE BEGIN 3 */
        HAL_GPIO_TogglePin(LED_RED_GPIO_Port,LED_RED_Pin);
        HAL_Delay(750);
        HAL_GPIO_TogglePin(LED_YELLOW_GPIO_Port,LED_YELLOW_Pin);
        HAL_Delay(500);
        HAL_GPIO_TogglePin(LED_GREEN_GPIO_Port,LED_GREEN_Pin);
        HAL_Delay(250);
} /* end of while */ *このかっこの前の行に挿入です
/* USER CODE END 3 */
```

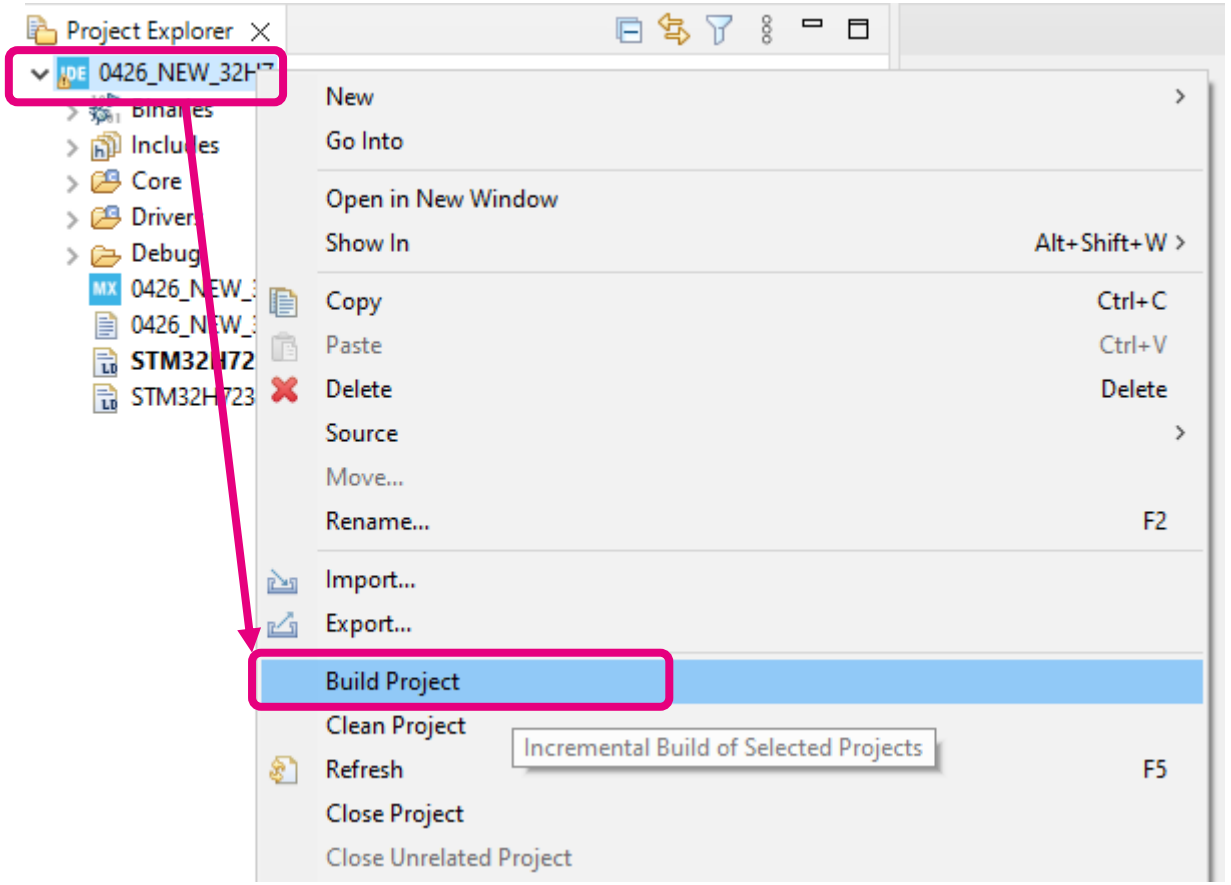
life.augmented

CubeIDE コード追記

追記が終わりましたらビルドします。

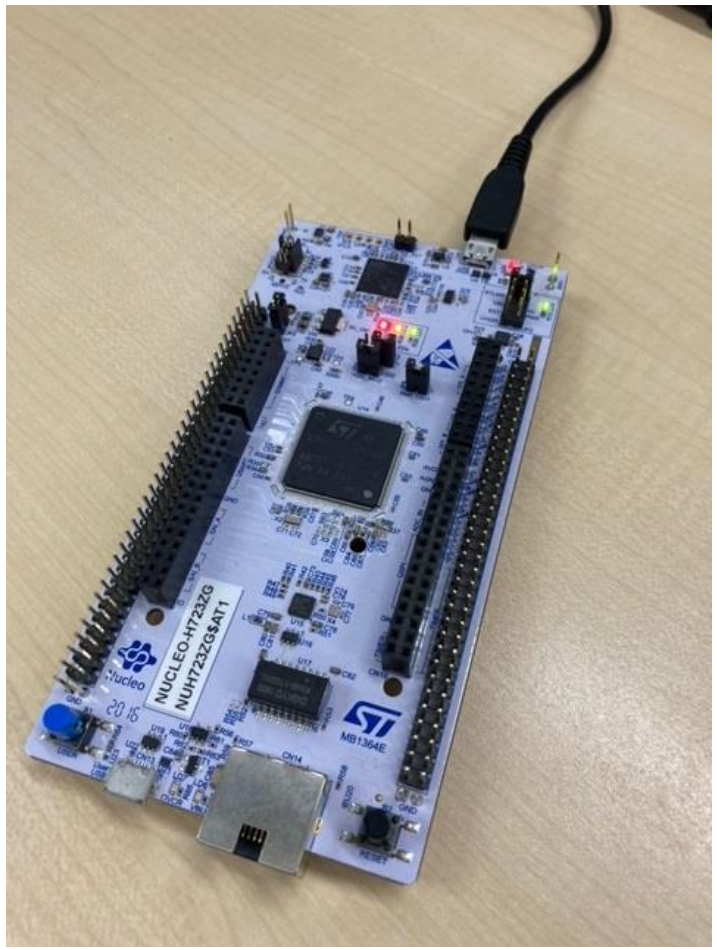
プロジェクトを右クリック → Build Project

右クリック



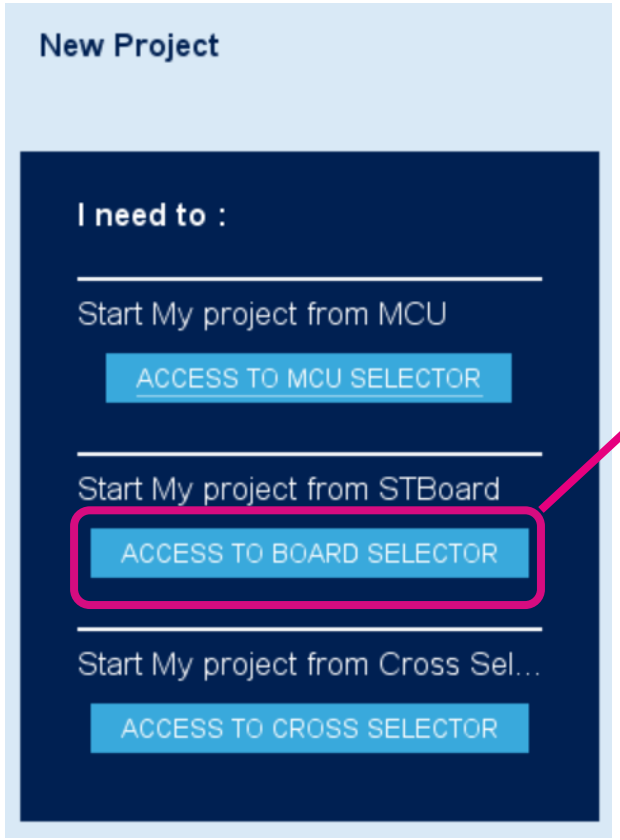
プログラム動作

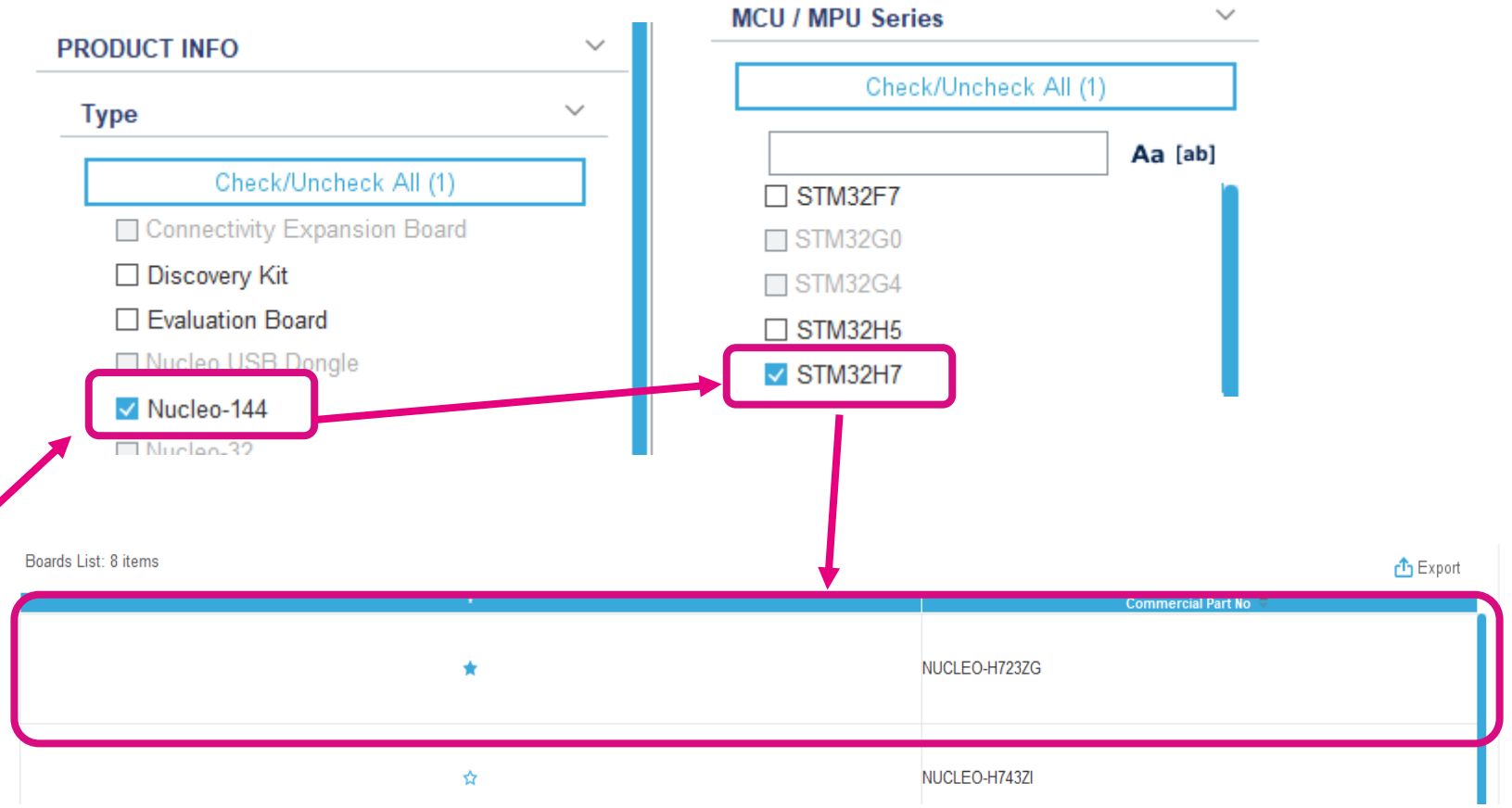
- プログラムをロードするとLEDが点滅します。



おまけ：Nucleoボードでのボード選択例

STM32CubeMXを起動した後に
評価ボードを選択することでプロジェクトを生成する事も出来ます。

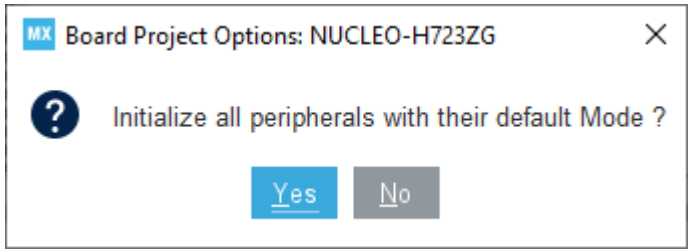




ACCESS TO BOARD SELECTORから
Type：**Nucleo-144** MCU Series：**STM32H7**
を選択し、リストから**NUCLEO-H723ZG**をダブルクリックします。

おまけ：Nucleoボードでのボード選択例

以下のダイアログではYesを押します。
これにより、ボードを使うときの標準的な設定が
自動的に適用されます。

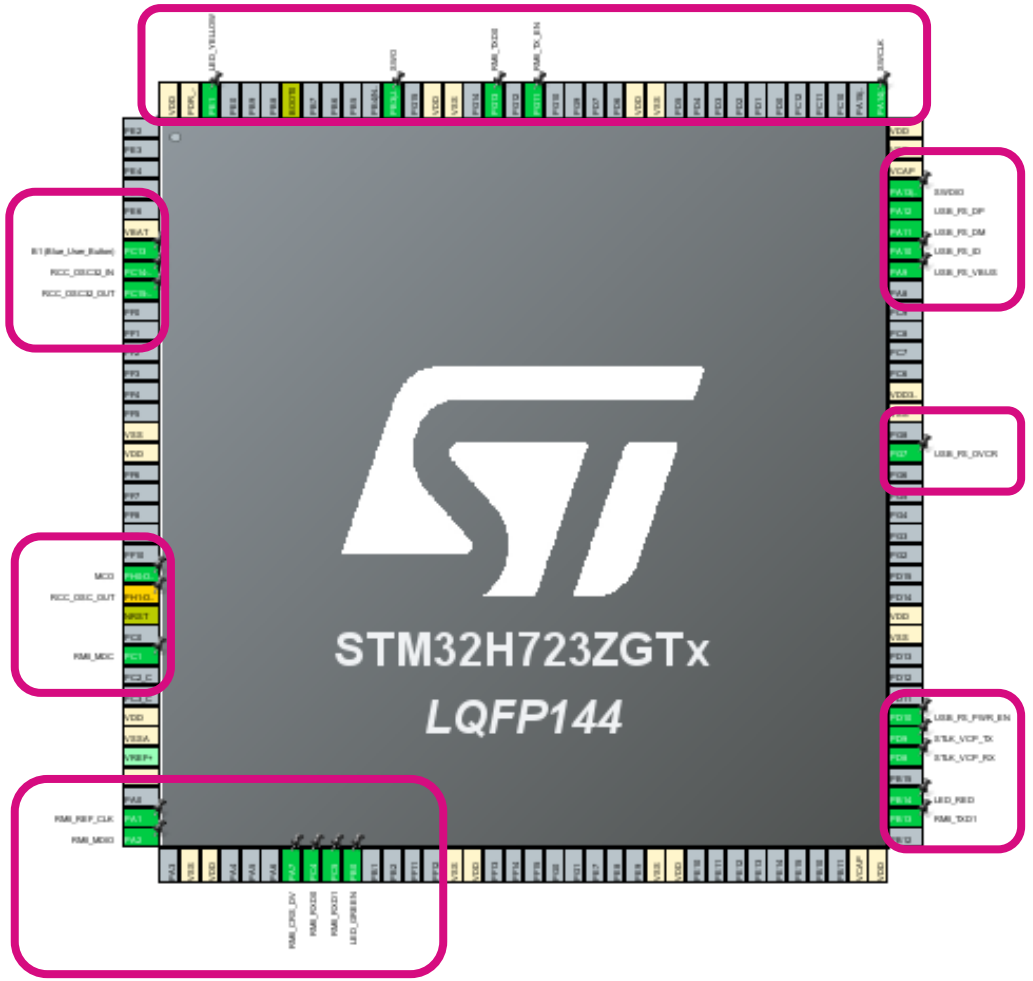


おまけ：Nucleoボードでのボード選択例

プロジェクト画面が開きます。

先の「I/O実践」の時に選んだMCUの選択時と異なり

ボード依存のピンは設定済みとなります。



life.augmented

参考資料

STM32CubeH7の展開場所

事前準備に以下の指示表示あり (STM32CubeMXセットアップ資料)

3. ユーザ名が日本語の方

ユーザ名が英数字+空白の方はこの作業は不要です。
メニューのHelp → Updater Settings のUpdater Settings タブで、Repository Folder に「C:\ST」を設定します。

ユーザ名が日本語の環境の場合

C:\ST

ユーザ名が英数字+空白の環境の場合：

C:\Users\<user name>\STM32Cube\Repository

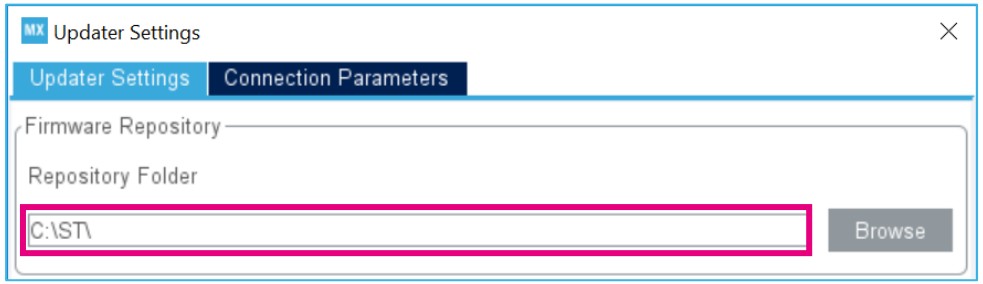
life.augmented

STM32CubeH7の展開場所

- STM32CubeMXを起動
- メニューより、Help → Updater Settings を選択
- Repository Folder 欄を確認

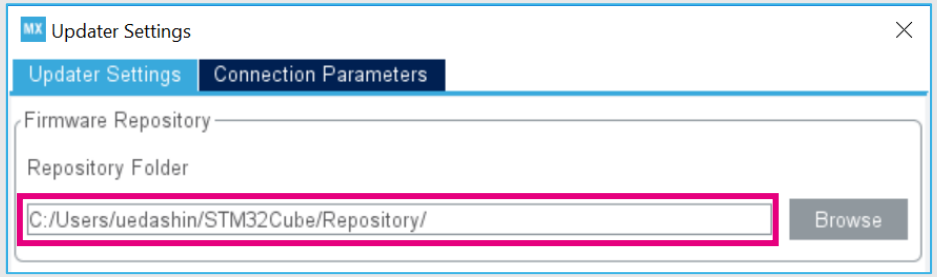
ユーザ名が日本語の環境の場合

C:\ST



ユーザ名が英数字+空白の環境の場合：

C:\Users\<user name>\STM32Cube\Repository

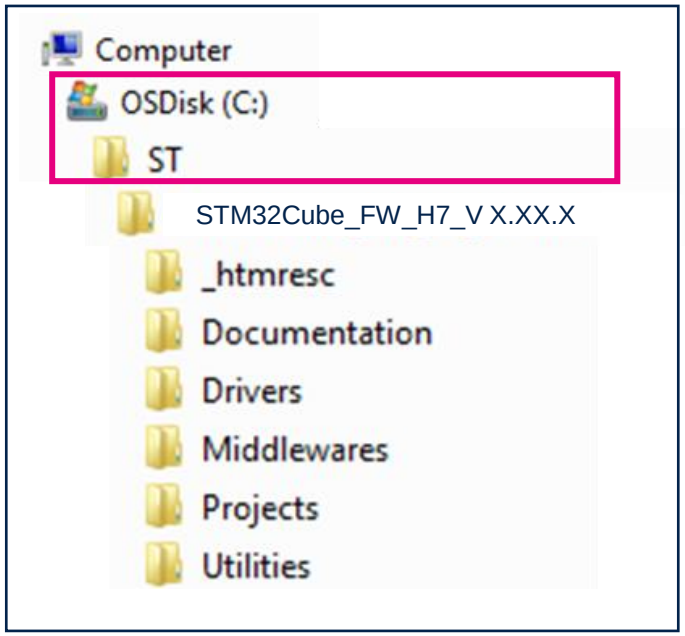


STM32CubeH7の展開場所

- STM32CubeH7の展開場所を確認

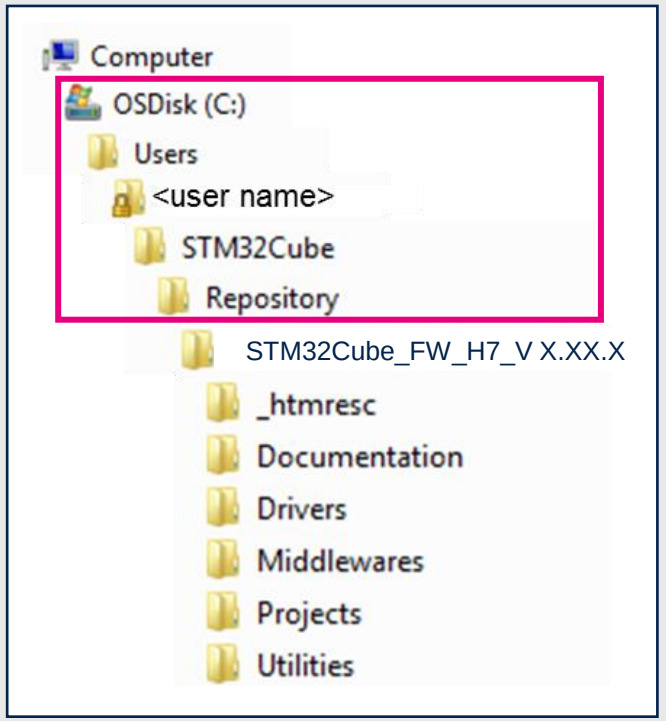
ユーザ名が日本語の環境の場合

C:\ST



ユーザ名が英数字+空白の環境の場合：

C:\Users\<user name>\STM32Cube\Repository



life.augmented

Thank you

life.augmented