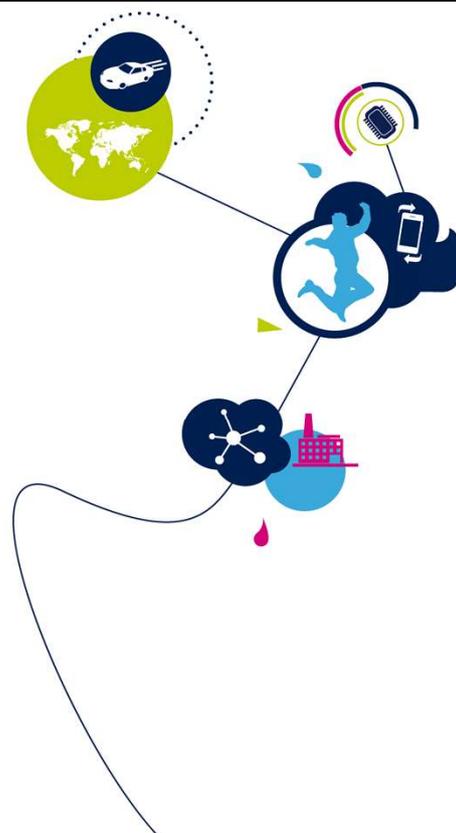


STM32G4 – STM32CubeMX

STM32CubeMXグラフィカル・ソフトウェア設定ツール

1.0版



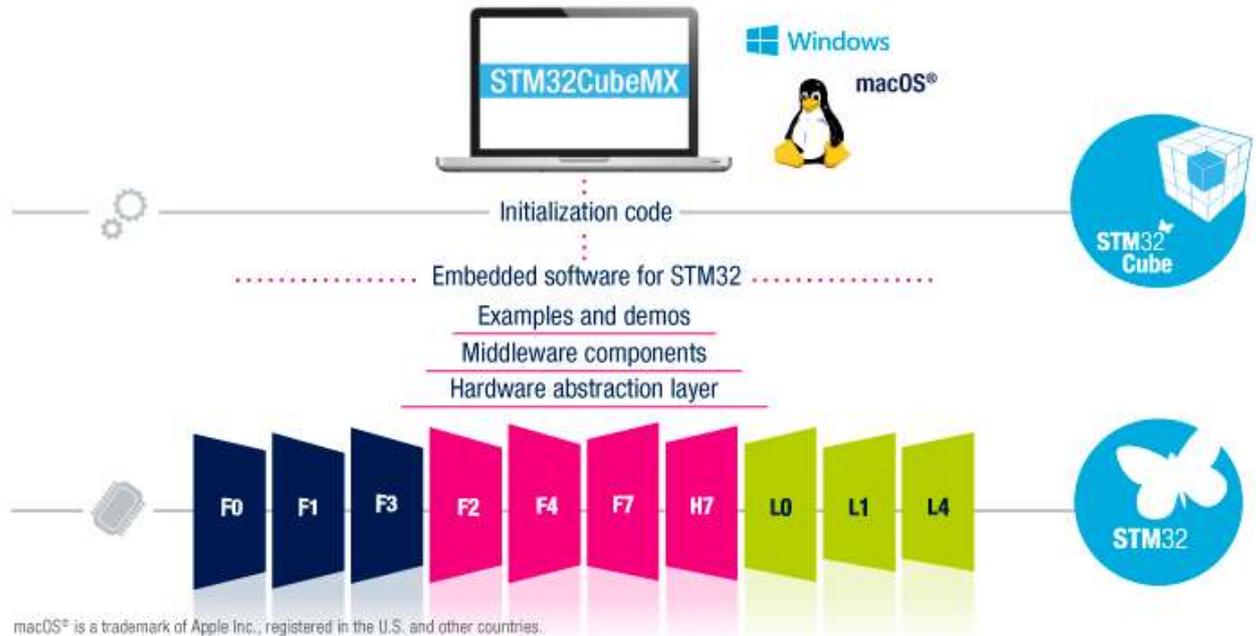
こんにちは。

STM32CubeMXコード生成ツールのプレゼンテーションへようこそ。

このツールの主な機能をカバーし、コードの設定と生成、コンパイルとデバッグ、およびSTM32ファミリのマイクロコントローラの消費電力を推定する機能を紹介します。

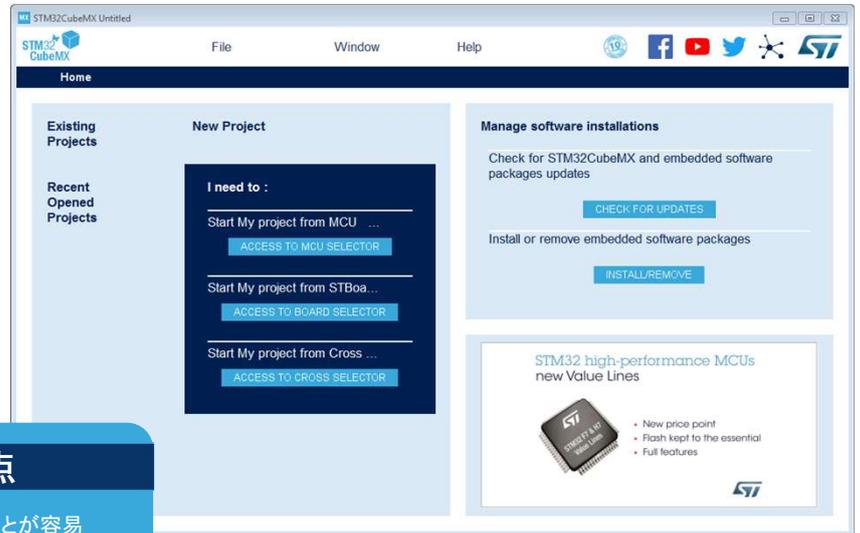
イントロダクション

2



この資料では特にSTM32G4シリーズのマイクロコントローラに関するプレゼンテーションですが、STM32CubeMXはSTM32ファミリ全体に共通のプラットフォームです。

- 理想的なマイコンを選択
詳細な設定が可能なツール
- ピンアウト
- クロックとオシレータ
- ペリフェラル
- 低消費電力モード
- ミドルウェア



このアプリケーションの利点

- 特定の目的に適したマイコンを選択することが容易
- 設計段階で利用可能なシミュレーション
- 開発スタート時のスピードを加速



STM32CubeMXアプリケーションは、ファームウェア、プロジェクトの初期構成をガイドするユーザー・インタフェースを介して、STM32マイクロコントローラを使用する開発者を支援します。ピンの割り当て、クロックツリー、統合されたペリフェラルを構成し、生成したプロジェクトの消費電力をシミュレートする手段を提供します。また、STM32マイクロコントローラの豊富なデータライブラリを使用します。このアプリケーションは、開発者がMCUの機能と消費電力に関して最適な製品を選択するのを助け、開発の初期段階を容易にすることを目的としています。

- ペリフェラルおよびミドルウェアのパラメータ
- 消費電力計算機能
- コード生成
 - ユーザ・コードをそのままに、コードの再生成が可能
- コマンドラインおよびバッチ操作オプション
- プラグインで拡張可能
- MCUセレクト
 - ファミリ、パッケージ、ペリフェラル、メモリ・サイズでフィルタリング
 - 類似製品を検索
- ピンアウト・コンフィグレーション
 - 使用するペリフェラルを選択、ピンに対しGPIOと代替機能を割り当て
- NVICとDMAの設定
- クロックツリーの初期化
 - オシレータを選択、PLLとクロック分周器の設定



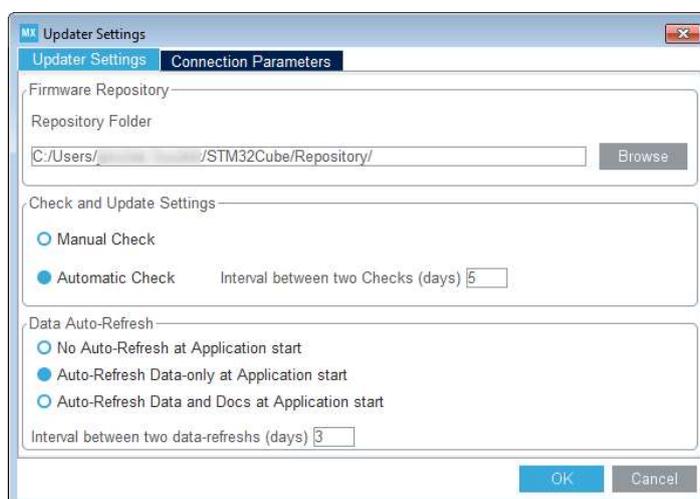
ユーザー・インタフェースは、適切なMCUを選択し、必要なペリフェラルを選択し、ピン構成を割り当てるというワークフローを中心に構築されています。

消費電力計算機能は、効率的なシステムの設計に役立ちます。最後に、プロジェクトの初期化コードを生成し、ユーザーコードをそのままに再生成することもできます。

インストール条件とセッティング

5

- STM32CubeMXはJava REのインストールが必要
 - リリースノートをチェックし、特定のバージョンや追加の要件を確認
 - マルチプラットフォーム・ツールはWindows、Linux、macOSで実行される
- インストール後、Alt+Sを押してGUIだけでなくCubeFWライブラリのアップデートの設定が必要
- SWライブラリの保存場所を選択



STM32CubeMXインストーラをSTウェブサイトから無料でダウンロードしてインストールします。

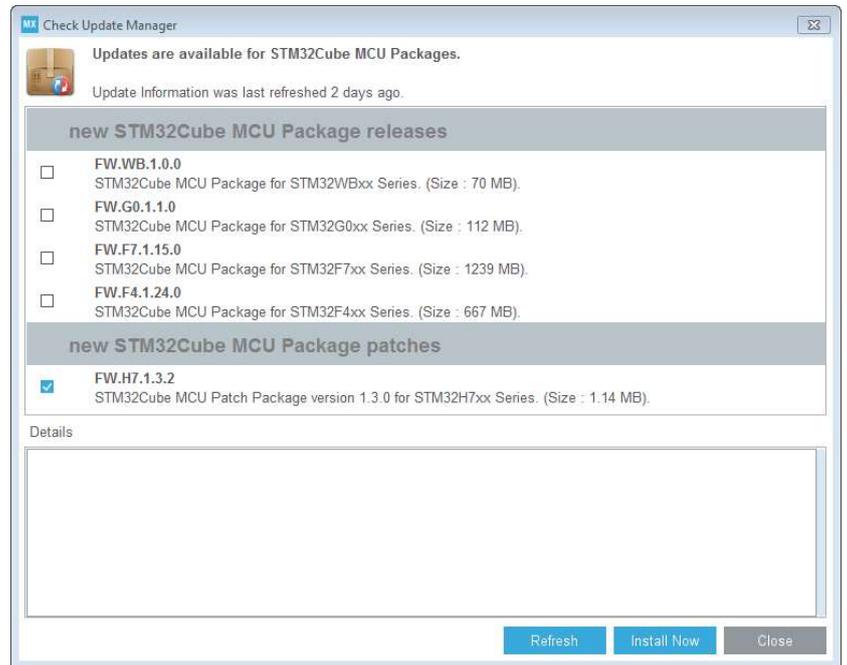
次に、[Settings]メニューで設定を行います。

- アップデータとライブラリのダウンロード用のメニュー(Alt +S)
- コード生成と開発ツールチェーンとの統合のためのメニュー(Alt+P)

注) ファームウェアを保存するレポジトリフォルダの保存場所は日本語が含まれるアドレスは使用できません

このセットアップが完了すると、新しいプロジェクトを作成することが出来ます。

- アップデータは[Help]メニューからアクセス
- ツールの新しいリリースと関連するCubeライブラリを検出
- ライブラリ・マネージャで新しいライブラリ・パッケージをダウンロード



インターネット接続が正しく接続されている場合、ツールそのものと、プロジェクトワークスペースの生成に使用されるコードライブラリを更新できます。

追加の STM32Cube ライブラリをダウンロードしたり、古いバージョンを取得したりするには、相互運用性の理由から「新規ライブラリのインストール」オプション (Alt+U) を使用します。

ただし、STM32CubeMX ツールは、すべてのバージョンのライブラリ間でテストされているわけではありません。

- マイコンの名前から検索...
 - 迅速にシリーズとラインで見つけることが可能
- ...もしくは、必要な機能から
 - パッケージ(ピン数)
 - RAMサイズ
 - 必要なメモリ・サイズ
 - ペリフェラル
 - コア、周波数
 - 価格
- ドキュメントへの便利なリンク
- テーブルをExcelファイルへエクスポート可能

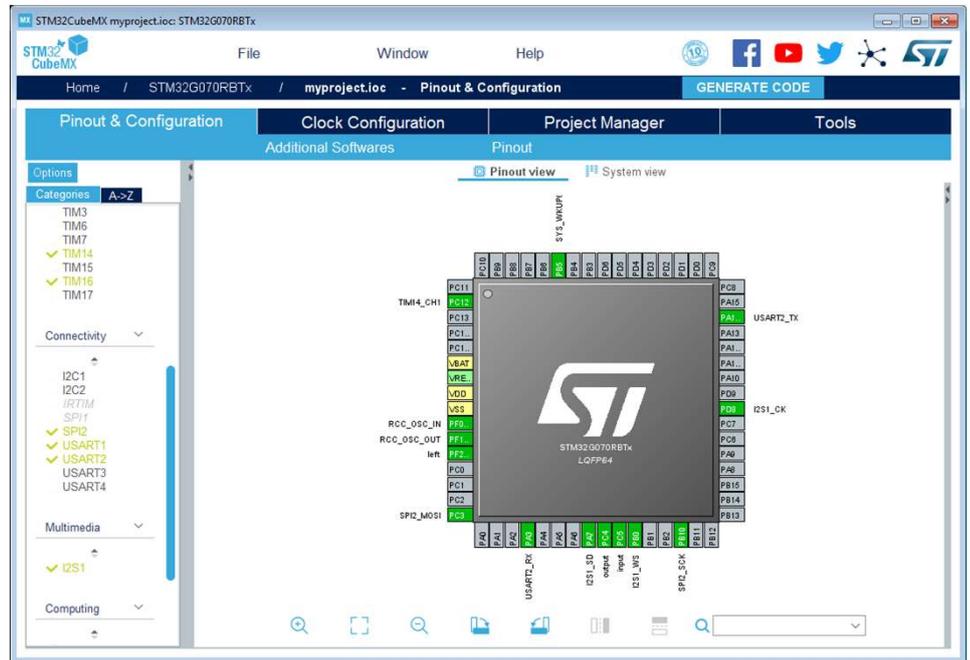


Part	Reference	M	Unit	Board	Package	Flash	RAM	IO	Free	GFX	TRNG	JIC
STM32F042K6	A. 0.9...			TSSOP20	32 k...	6 kBy...	16	48 M...	0.0	0	0	
STM32F042K6	A. 0.8...			UFOPFN28	16 k...	6 kBy...	23	48 M...	0.0	0	0	
STM32F042K6	A. 0.9...			UFOPFN28	32 k...	6 kBy...	23	48 M...	0.0	0	0	
STM32F042K6	A. 1.0...			LQFP32	16 k...	6 kBy...	26	48 M...	0.0	0	0	
STM32F042K6	A. 1.0...			UFOPFN32	16 k...	6 kBy...	28	48 M...	0.0	0	0	
STM32F042K6	Ac. 1.0...			NUCLEO	LQFP32	32 k...	6 kBy...	26	48 M...	0.0	0	
STM32F042K6	A. 1.0...			UFOPFN32	32 k...	6 kBy...	28	48 M...	0.0	0	0	
STM32F042K6	A. 1.0...			WLCSP36	32 k...	6 kBy...	30	48 M...	0.0	0	0	
STM32F042K6	A. 1.1			UFOPFN48	32 k...	6 kBy...	37	48 M...	0.0	0	0	
STM32F042K6	A. 0.9...			UFOPFN28	32 k...	6 kBy...	22	48 M...	0.0	0	0	
STM32F042K6	A. 1.0...			WLCSP36	32 k...	6 kBy...	29	48 M...	0.0	0	0	
STM32F042K6	A. 1.0...			LQFP48	16 k...	8 kBy...	39	48 M...	0.0	0	0	
STM32F042K6	A. 1.0...			UFOPFN48	16 k...	8 kBy...	39	48 M...	0.0	0	0	
STM32F042K6	A. 1.0...			LQFP48	32 k...	8 kBy...	39	48 M...	0.0	0	0	
STM32F042K6	A. 1.0...			UFOPFN48	32 k...	8 kBy...	39	48 M...	0.0	0	0	
STM32F042K6	A. 1.1...			LQFP48	64 k...	8 kBy...	39	48 M...	0.0	0	0	
STM32F042K6	A. 1.1...			UFOPFN48	64 k...	8 kBy...	39	48 M...	0.0	0	0	
STM32F042K6	A. 0.97			LQFP32	16 k...	8 kBy...	25	48 M...	0.0	0	0	
STM32F042K6	A. 0.97			UFOPFN32	16 k...	8 kBy...	27	48 M...	0.0	0	0	

MCUセレクト・ウィンドウは、「New Project」オプションを選択した後に表示されます。使用するMCUが決定されている場合は、フィルタ機能ですぐに検出されます。

まだ目的のMCUが決定していない場合は、アプリケーションの仕様、要求に基づいて、利用可能な製品をフィルタで選択できます。

- ピンアウトから:
 - ペリフェラルツリー
 - マニュアル
- 信号の再マッピングの自動化
- ペリフェラルやミドルウェア間の依存関係の管理 (FatFS, USB, AI ...)



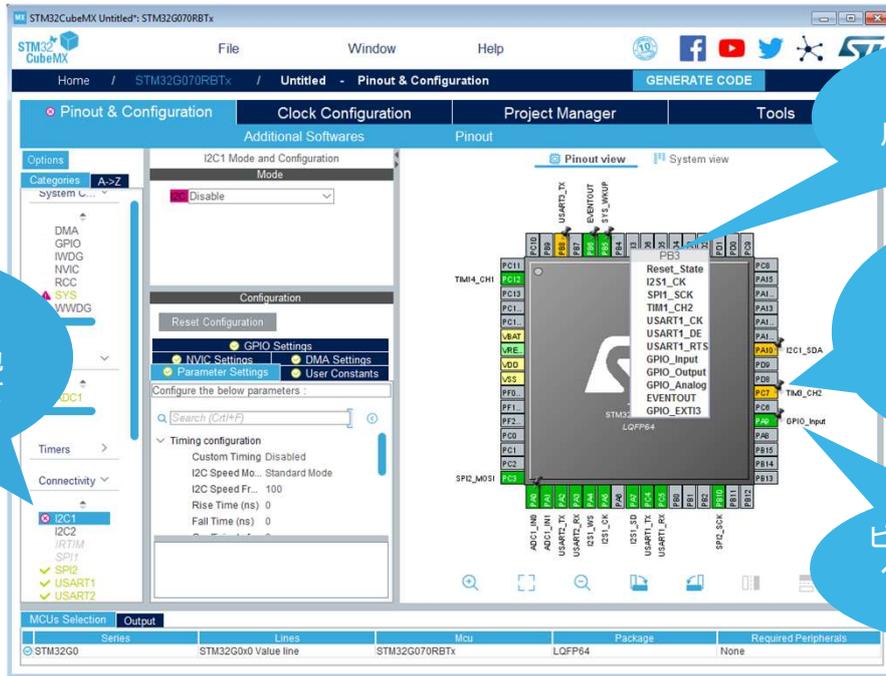
次のステップでは、使用するペリフェラルを選択し、該当する場合はピンの入出力を割り当てます。

独立したGPIOも構成できます。

信号はデフォルトのピンに割り当てられますが、別の位置に移動させることもできます。

たとえば、I2C1ペリフェラルが有効にした場合、ツールは自動的にデフォルトのピンにI2C1の機能を割り当てます。

このツールは、ペリフェラルと管理するソフトウェアコンポーネントとの間を自動的に結合します。



ペリフェラルは使用できず、その代替ピンはすべて他の場所に割り当てられる

ピンをクリックし、使用可能な機能を表示

オレンジはペリフェラルが有効になっておらず、ピンのみが割り当てられていることを意味する

ピンアイコンを設定し、ペリフェラルの配置を固定



代替機能用に予約されているピンが増えるほど、他のペリフェラルの残りの構成の選択肢は減少します。
 制限は、他のペリフェラルノードのアイコンの変更によって示されます。
 ピンを左クリックして、代替機能を表示します。
 ピンを右クリックして名前を付けるか、ピンの割り当てを選択します。
 特定のペリフェラルが有効になっていない状態でピンアウトを選択した場合、またはピンアウトに問題がある場合は、ピンが緑色ではなくオレンジ色に変わります。

- ペリフェラル設定の可能な状態について
 - 淡色表示: 別のモードが設定されているため、そのペリフェラルは使用できない
 - 黄: そのペリフェラルは制限付きで使用可能
 - 赤: そのペリフェラルに必要な信号はピンにマッピングできない状態
- 信号はピンマッピング・ビュー上から直接設定、移動可能
 - ピンをクリックし、使用可能なペリフェラルのリストを表示、選択
 - そのペリフェラルにおいて他に設定できるピンを表示するには、ピンをCtrlキーを押しながらクリック。他の場所にドラッグ & ドロップも可能
 - 未設定のピンは、コード・ジェネレータが省電力アナログ・モードに設定



ペリフェラルモードには、さまざまな状態があります。

淡色表示: 別のモードを設定する必要があるため、このペリフェラルは使用できません。マウスポインタを淡色表示モードに置くと、クロックソースが無効になっているか、他のペリフェラルの依存関係が必要になる場合もあります。

黄: 一部のオプションで競合によってブロックされるため、モードは制限付きで使用できます。たとえば、選択可能なクロックピンがすべて取られるため、USART が同期モードに設定されていない場合などがあります。

赤: このモードに必要な信号はピン配列にマッピングできません。これは、重要な信号に他のペリフェラルが使用する代替ピンがすべて含まれた場合などに発生する可能性があります。

信号はピンアウトビューから直接設定、または移動できます。

ピンをクリックして、可能な信号のリストを表示し、1つを選択します。これは、ペリフェラルが割り当てられていない GPIO に対して機能します。

信号の代替ピンを表示するには、Ctrl キーを押しながら信号をクリックします。その後、(Ctrl キーを押しながら)新しいピンに信号をドラッグ&ドロップできます。

未使用のピンをすべてアナログに手動で設定する必要はありません。これを行う半自動化されたステップが、別のウィンドウで設定できます。

- 設定済みペリフェラルとミドルウェアのグローバル・ビュー
- 設定エラーのハイライト
 - + 設定していない
 - ✓ OK
 - ⚠ ブロックされない問題あり
 - ✗ エラー



GPIO の構成は正しくないが、コードが生成される可能性がある

クリックしてDMAを設定

ピンアウトとシステム・ビューをすばやく切替え

詳細設定はここをクリック

コンフィグレーション・エラー:コード・ジェネレータは警告メッセージを表示

メインウィンドウの「Configuration」タブには、STM32CubeMXのセットアップに役立つ、構成可能なすべてのハードウェアおよびソフトウェア・コンポーネントの概要が表示されます。

構成オプションにアクセスできる各ボタンには、構成状態を示す小さなアイコンが表示されます。

既定の状態は構成されていません。ペリフェラルまたはミドルウェアのボタンをクリックすると、その設定オプションが表示されます。

すでに構成されている場合でも、さらなる変更が可能です。

警告サインは、不適切な構成に関する通知をし、この状態でコードが生成された場合、ペリフェラルは機能しません。

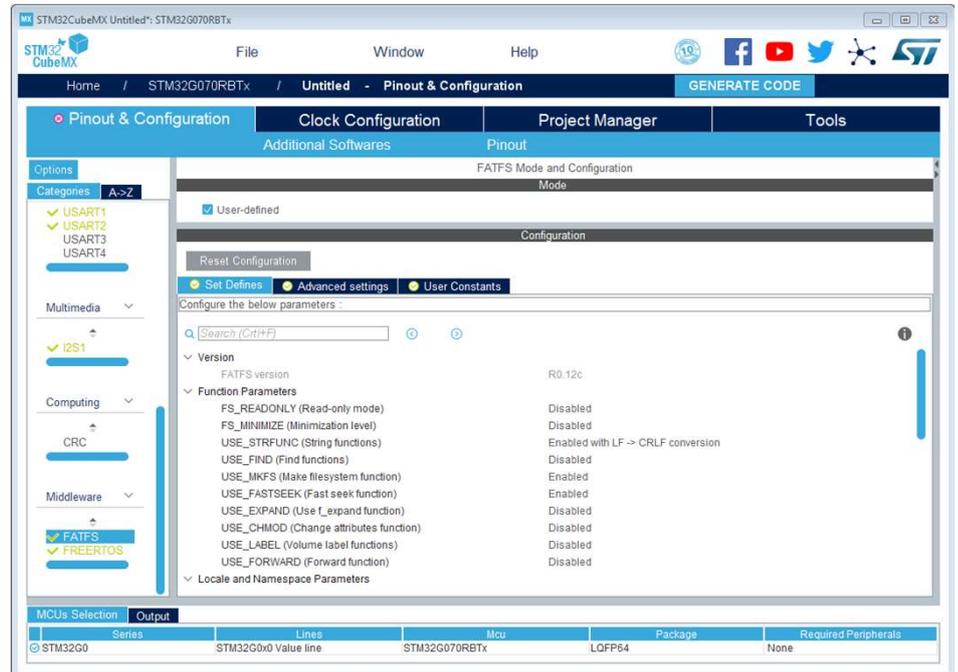
重大なエラーは赤い“X”で表され、続行するには構成を変更する必要があります。

ペリフェラルやコンポーネントを追加するには、[Pinout & configuration] タブに戻ります。



life.augmented

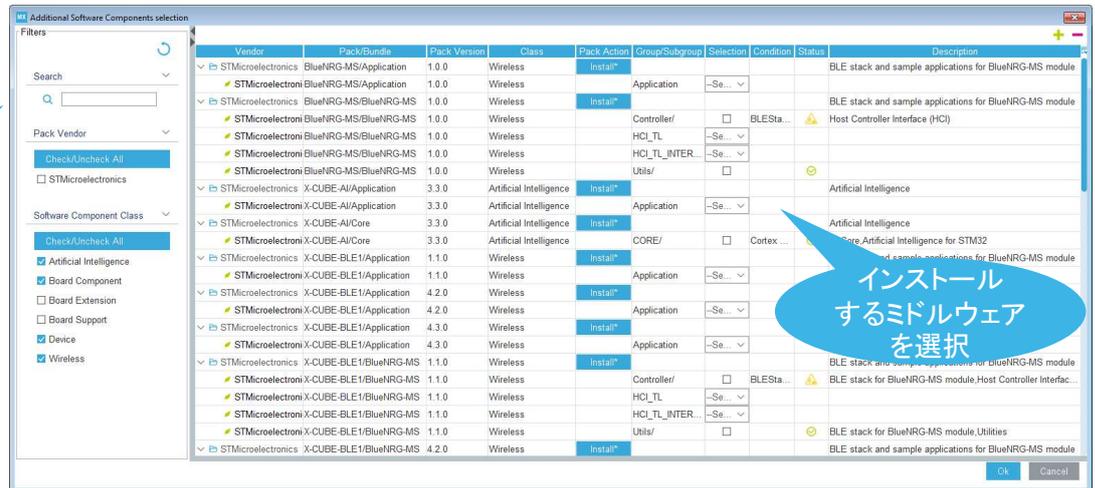
- サポートされている各ソフトウェア・コンポーネントに固有のオプションを表示
- すべての設定は、論理グループに編成
- 説明と制約は、クイックリファレンスに使用可能



ミドルウェアソフトウェアコンポーネントには、それぞれ異なるオプションがありますが、すべて同様の方法で提示され、初期化オプションに簡単にアクセスでき、有益な説明を提供します。

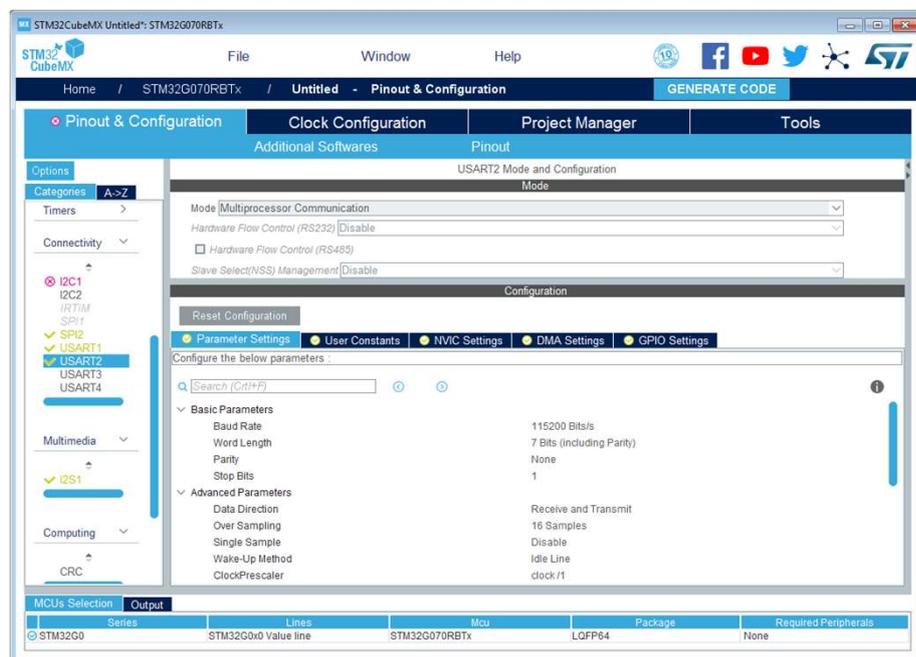
ミドルウェア・パッケージの追加

- ほとんどのミドルウェア・パックは基本パッケージに含まれていないが、オプションでダウンロードし追加可能



STM32で利用可能な膨大な量のミドルウェアパックは、単一のパッケージとして配布するのは現実的ではありません。
[Pinout & configuration] タブの [Additional Software] リンクをクリックすると、選択した MCU で利用可能なほとんどのミドルウェアパッケージが見つかります。

- 使用可能なすべての初期化パラメータに、簡単な説明とオプションが表示されている
- 割り込みはペリフェラルに割当て可能
- DMAは、該当する場合に関連付けられる
- 入力および出力を備えたGPIO



ペリフェラルを設定する場合、ダイアログ ウィンドウには基本的なパラメータ、依存関係、および制約が表示されます。設定が必要な項目についてはシンプルなドロップダウンメニューが使用されます。

割り込み優先順位は「NVIC settings」タブでのみ設定できます。ペリフェラルウィンドウは、各割り込みを有効または無効にする場合にのみ使用できます。

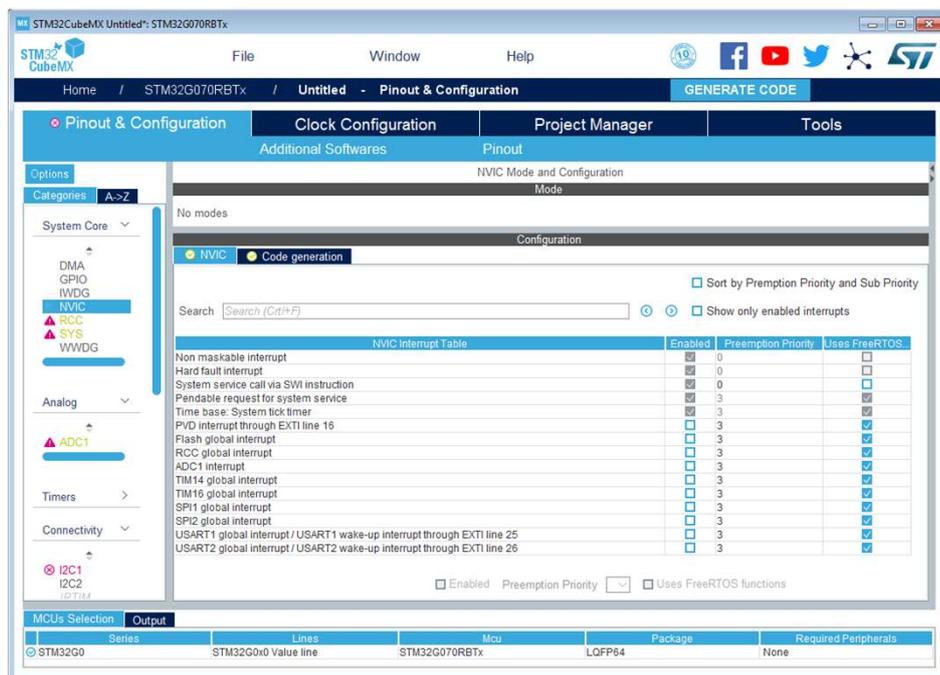
[DMA settings] タブには、初期化に関連する DMA 要求のすべてのパラメータが含まれていますが、実行時パラメータ (開始アドレスなど) はここで管理されません。

[GPIO settings] タブは、GPIOのパラメータと機能、ピンフィルタリング、および容易に識別を可能にするために各信号にラベル付けを定義するために使用されます。

NVICの設定パネル

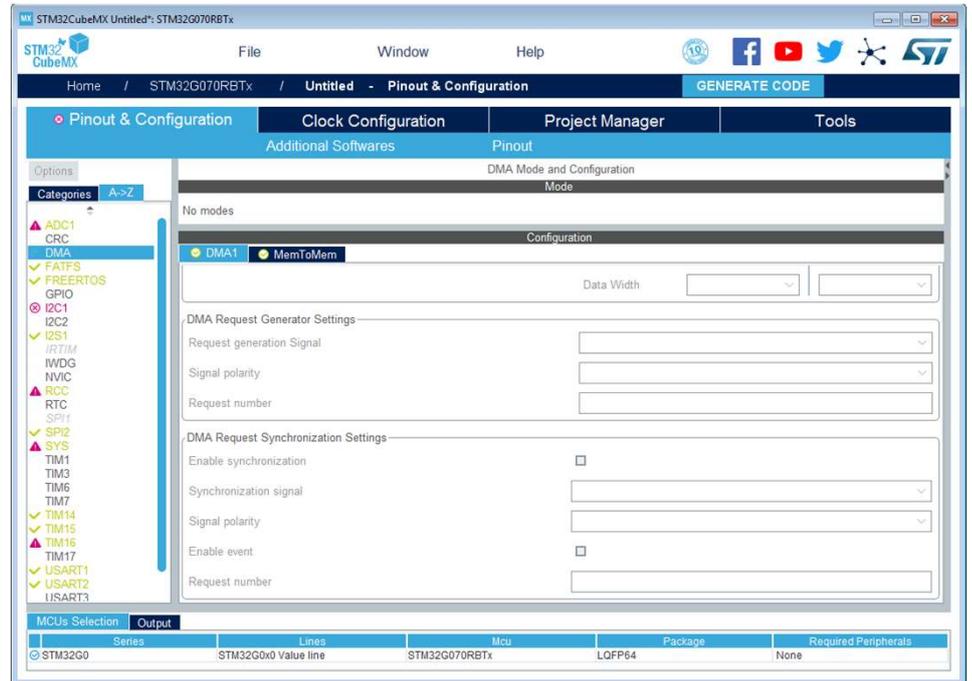
16

- すべての割込みのための単一のコントロール・パネル
- 優先順位とサブ優先順位を管理
- リスト内の割込みの検索、フィルタ処理、並べ替え
- [コード生成]タブで、割込みの初期化をカスタマイズ可能



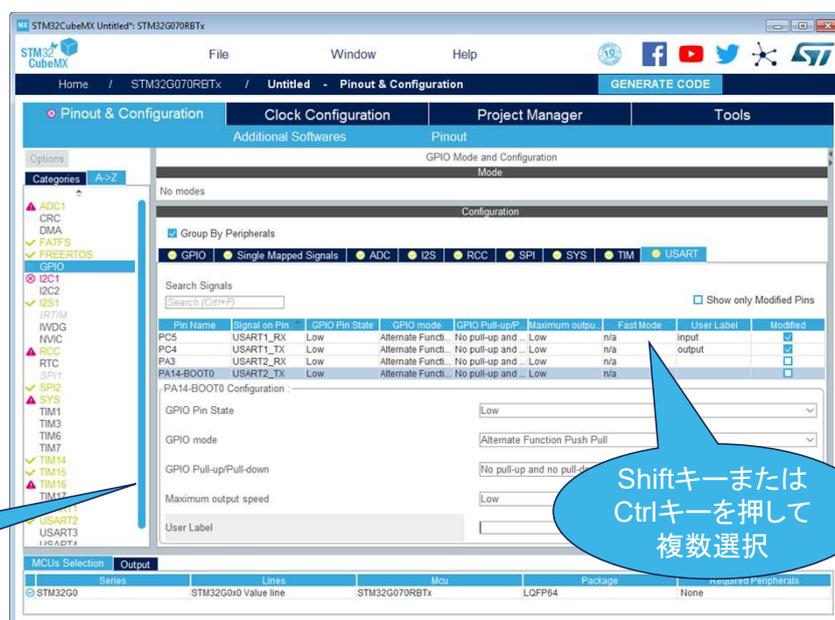
使用可能な割り込みと有効な割り込みの概要とその優先順位が一度に表示される分かりやすい画面は、STM32CubeMX のもう1つの利点です。このウィンドウは、選択したペリフェラルの割り込みを有効にし、優先順位を構成するために使用されます。

- メモリからメモリを含むすべてのDMA要求を管理
- 方向、優先度、その他の設定を構成



対応する DMA チャンネルのタブを選択し、指定されたペリフェラルの DMA を追加するには、「Add」ボタンをクリックします。すべての構成オプションを確認してください。これは DMA チャンネルを構成しますが、DMA 転送を完全には記述しません。これは、アプリケーション上で行う必要があります。

- アプリケーションにとって、GPIOパラメータは実用的な設定が必要
- デフォルトでは、低速およびプルアップなしとして選択される
- 複数のピンを選択し、同じ構成に設定可能



速度が正しいことを確認

ShiftキーまたはCtrlキーを押して複数選択



life.augmented

ピン設定ウィンドウの GPIOタブは、各ピンの構成と初期化設定を容易にします。

各ピンは、ピン構成の概要とユーザーラベルを示すテーブル形式でリストされています。

ドロップダウンメニューを使用して、選択したピンの並べ替え、検索、変更が可能です。

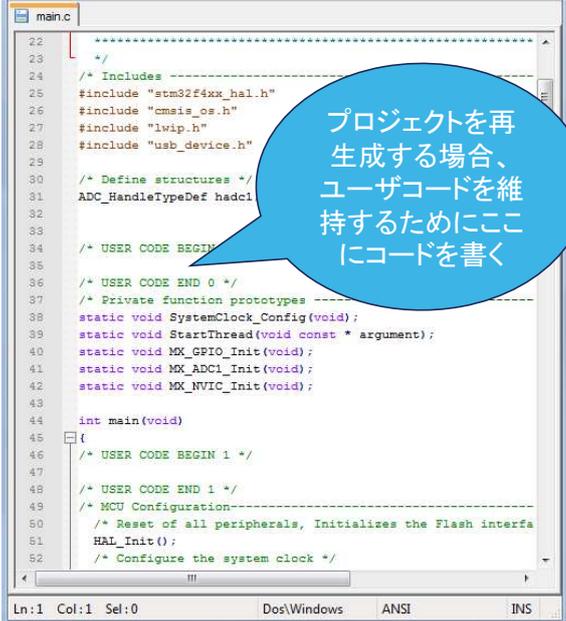
ツールによって割り当てられた既定値では、特定のペリフェラル構成では機能しない場合があります。

ツールで選択した GPIO 速度がペリフェラルの通信速度に対して十分であることの確認や、必要に応じて内部プルアップ等の設定が可能です。

設定を素早く割り当てるには、ピンを個別に設定するのではなく、ピンのグループを選択してみてください。タブを使用して、特定のペリフェラル専用のピングループを取得します。

初期化中に適用される設定はユーザープログラム実行中に変更できませんが、それはSTM32CubeMX ツールのサポート範囲外であることを注意してください。

- すべての初期化コードをC言語で生成
- サポートされている開発ツールチェーンのプロジェクト・ファイルを生成
- ユーザーコードは専用セクションに追加でき、再生成時に保持される
- 最新のライブラリバージョンを使用するか、再生成しても同じままにするオプションがある



```
22 .....
23 */
24 /* Includes .....
25 #include "stm32f4xx_hal.h"
26 #include "cmsis_os.h"
27 #include "lwip.h"
28 #include "usb_device.h"
29 .....
30 /* Define structures */
31 ADC_HandleTypeDef hadcl
32 .....
33 .....
34 /* USER CODE BEGIN .....
35 .....
36 /* USER CODE END 0 */
37 /* Private function prototypes .....
38 static void SystemClock_Config(void);
39 static void StartThread(void const * argument);
40 static void MX_GPIO_Init(void);
41 static void MX_ADC1_Init(void);
42 static void MX_NVIC_Init(void);
43 .....
44 int main(void)
45 {
46 /* USER CODE BEGIN 1 */
47 .....
48 /* USER CODE END 1 */
49 /* MCU Configuration.....
50 /* Reset of all peripherals, Initializes the Flash interfa
51 HAL_Init();
52 /* Configure the system clock */
```

すべての入力、出力、およびペリフェラルを構成すると、コードを生成する準備が整います。

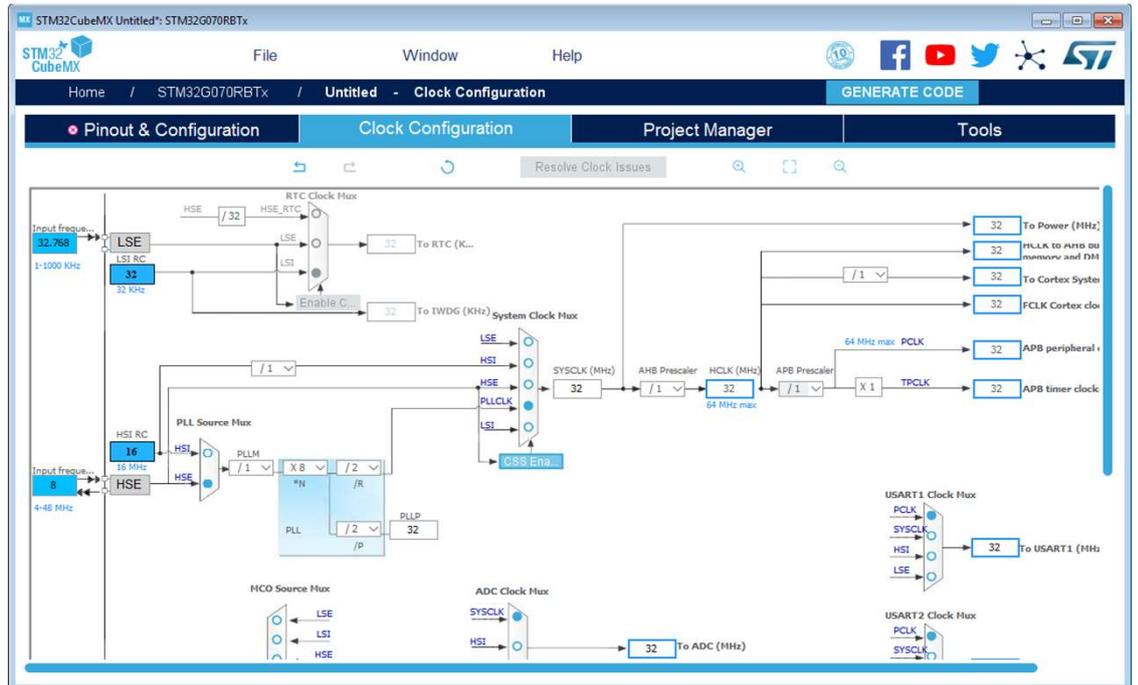
まず、メインウィンドウの「プロジェクト」メニューで設定を確認します。

Keil、IAR、および CubeIDE のツールチェーン向けに、プロジェクトを生成するためサポートされているいくつかの開発ツールの1つを選択します。

ユーザーが後から追加したカスタムコードに影響を与えずに、STM32CubeMX を使用して初期化設定をさらに変更するには、ユーザーコードを“USER CODE”コメントブロックの制約の間に記述する必要があります。

このオプションを有効にする方法については、この後のスライドで説明します。

- すべてのクロック値の即時表示
- アクティブなクロックパスと非アクティブなクロックパスは区別
- クロック制約と機能の管理



クロックコンフィグレーションタブには、クロックパスの概略図と、すべてのクロックソース、逡倍回路、乗数が表示され、実際のクロックの速度が表示されます。

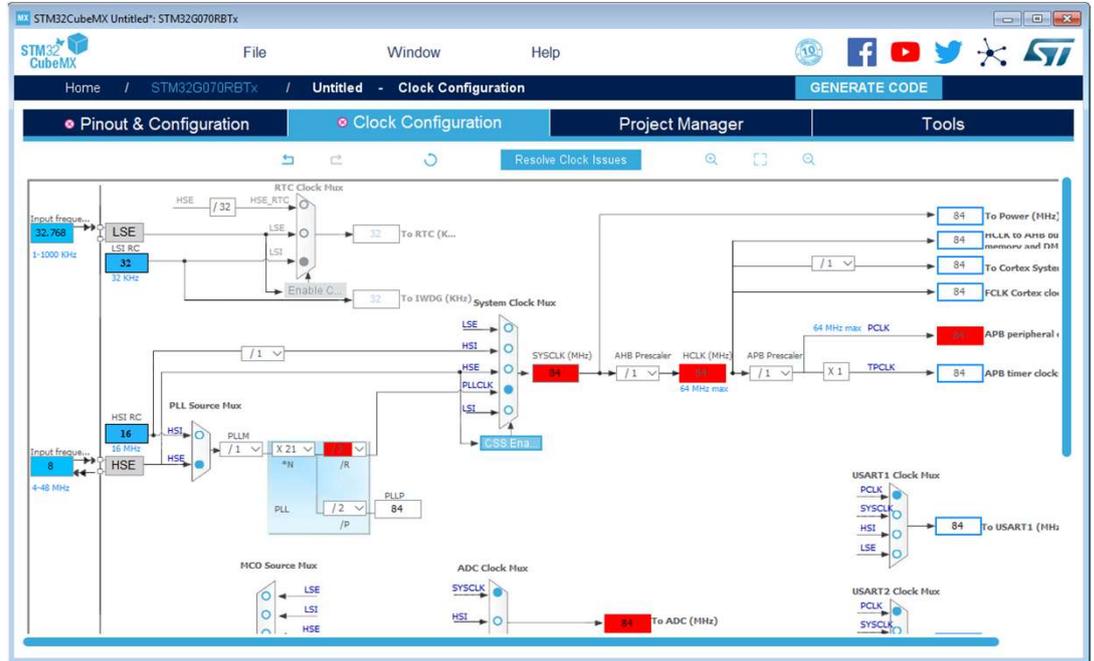
アクティブなクロック信号と有効なクロック信号は青色でハイライト表示されます。

ドロップダウンメニューとボタンは、実際のクロック構成を変更するのに役立ちます。

クロック・コンフィグレーション(続き)

21

- すべてのクロック値の即時表示
- アクティブなクロックパスと非アクティブなクロックパスは区別
- クロック制約と機能の管理

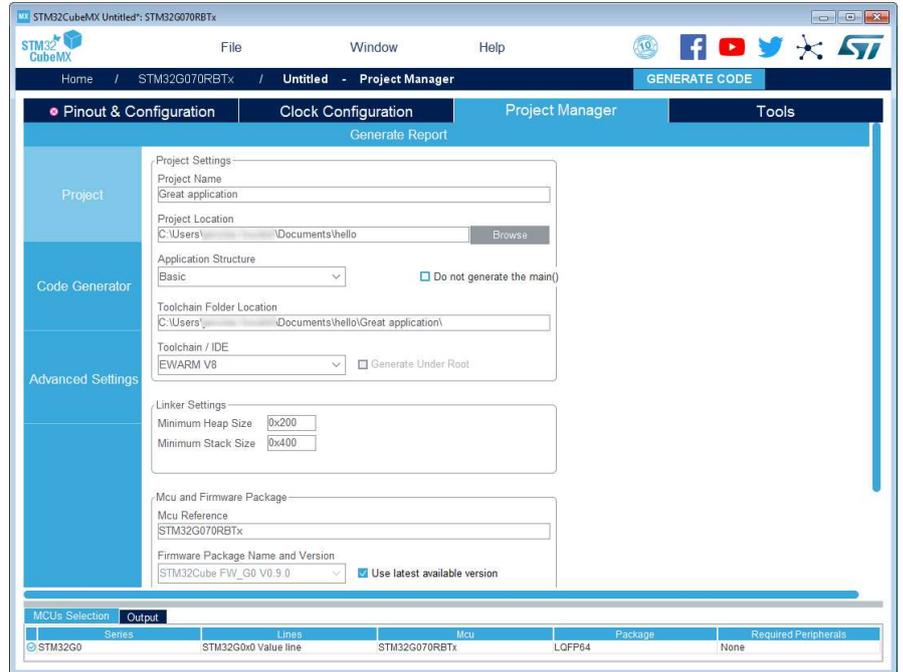


クロックコンフィグレーションタブには、クロックパスの概略図と、すべてのクロックソース、逓倍回路、乗数が表示され、実際のクロックの速度が表示されます。

アクティブなクロック信号と有効なクロック信号は青色でハイライト表示されます。

ドロップダウンメニューとボタンは、実際のクロック構成を変更するのに役立ちます。

- 保存時にプロジェクトに名前を付ける
- プロジェクトの保存場所を指定
- 優先するツールチェーンを選択
- MCUタイプと使用するライブラリのバージョンを確認



このウィンドウは、プロジェクトを保存するときに使用できます (Save as..)。toolchain フォルダは、実際のツールチェーン アプリケーションの場所ではなく、ツールチェーンのワークスペースを配置する場所を指します。このダイアログウィンドウの限定バージョンは、プロジェクト設定を表示する Alt + P ショートカットを使用して使用することもできます。

ライブラリ・パッケージ

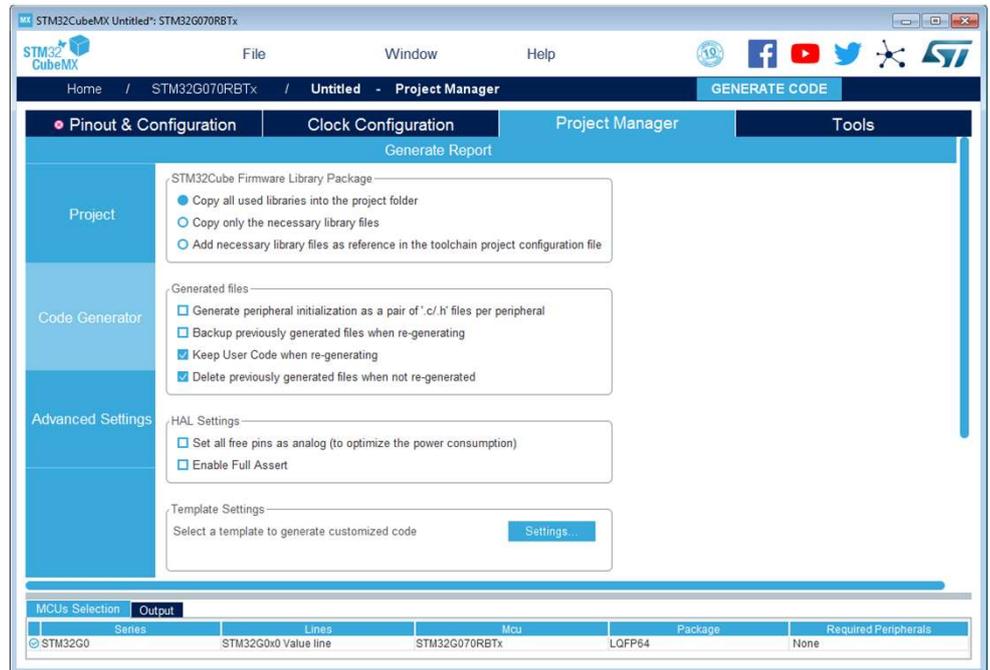
- ライブラリ全体または必要な部分は、生成されたプロジェクト・フォルダにコピーされる
- または、ライブラリを元の場所に保管し、すべてのプロジェクトから参照

生成されたファイル

- 各ペリフェラルは、個別のファイルまたは共通ソース・ファイルで初期化
- 古いファイルを操作するためのオプション
- ユーザコードをそのまま保持するオプション

HAL設定

- 使用可能なピンをアナログに設定すると消費電力が削減されるが、設定した場合はピン・マッピング上でSWD/JTAGを明示的に選択するように注意
- 完全なアサートはデバッグに役立つ。



STM32CubeHALライブラリは、さまざまな方法でプロジェクトに関連付けられます。プロジェクトをコンパクトパッケージとして移行する場合、またはライブラリコードをカスタマイズする必要がある場合は、[copy] オプションを選択します。ライブラリを元の場所に保持しておくと、複数のプロジェクト間でライブラリの最新バージョンを簡単に共有できます。

また、`stm32G4xxx_hal_msp.c` ファイル内のすべてのペリフェラルの初期化コードを生成したり、ペリフェラルごとに1つのファイルとして生成したりすることもできます。

古いファイルをバックアップまたは削除するオプションは、推奨されるワークフローです。オプションは再生成機能に関連付けられていることを覚えておいてください。ここでも、“再生成時にユーザーコードを保持する”オプションが有効になります。

「Set all free pins as analog」設定は消費電力を抑えるのに役立ちますが、SWD/JTAGインタフェースがピンアウトタブで選択されていない場合、このオプションはデバッグインタフェースを無効にします。

“完全なアサート”はHAL関数に渡されるパラメータをチェックすることを可能にし、過度のデバッグ作業を行わずにユーザーコード内のバグを明らかにするのに役立ちます。

- STM32ファミリ全体に向けた普遍的な取り組みにより、ツールが特定の製品の特定の機能に焦点を当てることができなくなることがあります。
- **STM32CubeMX GUIツールは、リファレンスマニュアルやデータシートに代わるものではありません。**
 - 詳細については、常に技術文書を参照してください。
 - 重要な機能は製品やHALでは利用できてもGUIでは利用できないことがあります。
- GUIはプロジェクトの開始と作業設定の初期化に役立ちますが、設定は実行時に動的に変更することもできません (GPIO、NVIC優先度、クロック設定など)。



ユーザーインターフェイスは優れたツールであり、すべてのSTM32マイクロコントローラーのユニバーサルアシスタントです。ただし、多様化したSTM32ポートフォリオの有用な概要を提供しながら、各製品のすべての詳細に取り組むことはできません。

疑問がある場合は、より詳細で正確な情報について、リファレンスマニュアルまたはデータシートを参照してください。詳細については、アプリケーションノートとサンプルをお読みください。

STM32CubeMXを使用してアプリケーションを起動し、動作するプロトタイプをすばやく取得してから、動的な変更が必要な場合にコードを変更するのが一般的な方法です (通常、同じアプリケーションで別のクロックまたは新しいGPIO構成をサポートするため)。

ユーザーがSTM32CubeMXジェネレーターによって定義されたユーザー領域内にコードを記述した場合、ユーザーインターフェイスのトップレベルでいくつかの変更が必要な場合は、最初のSTM32CubeMXセットアップに戻すことができます。これには通常、GPIOピン構成の追加、別のクロックの選択、NVIC優先度の変更などが含まれます。



消費電力 / 計算機能 / プラグイン



組み込みアプリケーションを開発する場合、低消費電力が主な設計目標であることがよくあります。データシートから電力消費レベルを抽出するのは大変な作業となります。消費電力計算ツールは、データシート値を抽出し構成可能なシナリオから有益な見積もりを生成することで、タスクを簡略化します。

- 消費電力計算機能(Power Consumption Calculator : PCC)は、データベースからの一般的な値を使用し、STM32MCUの消費電力、DMIPS、およびバッテリーの寿命を推定
- STM32CubeMXに統合されたGUIツール
- 妥当性チェックを使用し、構成可能なシナリオを使用
- バッテリー・セレクタを使用するか、カスタム・バッテリーを定義
- 他のMCUや他の電源オプションとの比較が容易
- レポートのインポート、エクスポート、生成



消費電力計算機能は、メインまたは補助電源として使用されるバッテリー寿命を推定するためにも使用できます。シーケンスは簡単にインポートおよびエクスポートできます。不正な状態遷移も検出されます。2つの異なるMCUのシーケンス実行を比較してレポートを生成することもできます。

PCC 設定パネル

シーケンス構成

結果の概要

Step	Mode	V _{DD}	Range/Scale	Memory	CPU/Bu	Clock Config	Peripherals	Step Current	Duration
1	RUN	1.8	Range1-High	FLASH	16 MHz	HSE BYP	ADC1fs_10_ksp...	2.55 mA	1 ms
2	STANDBY	1.8	NoRange	FLASH	16 MHz	HSI	RTC*	770 nA	1 ms
3	LOWPOWER_RUN	1.8	NoRange	FLASH	2 MHz	HSI Regulator_LP	GPIOA GPIOB GPIO...	496.73 µA	1 ms
4	LOWPOWER_SLEEP	1.8	NoRange	Flash-Po...	1 MHz	HSI Regulator_LP	IOPORT_Bus RTC...	243.3 µA	?
5	VBAT	3.0	NoRange	FLASH	16 MHz	HSI Regulator_LP	RTC*	2 nA	1 ms
6	RUN	1.8	Range2-Medium	SRAM1 Fl.	4 MHz	HSE BYP	GPIOA GPIOB GPIO...	559.26 µA	1300 µs
7	SLEEP	1.8	Range2-Medium	FLASH	8 MHz	HSE BYP	GPIOA GPIOB GPIO...	489.02 µA	2 ms
8	STOP0	3.0	NoRange	Flash-Po...	16 MHz	HSI_KERON	IWDG* RTC*	308 µA	500 µs

Consumption Profile by Step

Sequence Time / Ta Max: 11 ms / 129.7 °C
 Battery Life Estimation: 1 year, 7 months, 7 days, 11 hours
 Average Consumption: 624.37 µA
 Average DMIPS: 11.93 DMIPS

消費電力計算機能は、STM32CubeMXメインウィンドウの4番目のタブにあります。ウィンドウはさらに複数のウィンドウに分割されています。

一般設定ウィンドウには、一般的な動作条件と現在選択されている MCUが表示されます。

2番目のウィンドウには、シミュレーションシーケンスとそのコントロールが表示されます。

シミュレーションを実行するボタンはありません。結果は即座に表示されます。

- 使用するMCUはSTM32CubeMXから取得
 - データシートへの直接リンクを使用し、より詳細な情報を取得可能
- パラメータの選択
 - 温度および電圧の選択は、選択したMCUに応じて制限される場合がある
- バッテリーの選択 - 既存または独自の定義
 - バッテリーは容量、電圧、自己放電、電流制限によって定義
- インフォメーションノート
 - 見積の制限について警告が目的

STM32G070RBTx

Series	STM32G0
Line	STM32G0x0 Val...
Datasheet	DS12766_Rev0

T_A 25°C / V_{DD} 1.8V

T _{Ambient}	25°C
V _{DD}	1.8

Li-MnO2(CR2477) (1x1)

In Series	1	In Parallel	1
Capacity	850.0 mAh		
Self Discharge	0.12 %/month		
Nominal Voltage	3.0 V		
Max Cont Current	2.0 mA		

[Information Notes](#) >

[Help](#) >



一般的な PCC 設定ウィンドウは、ほとんどが有益で、選択した MCU とデフォルトの電源を要約します。選択したMCUと利用可能な電力消費データに応じて、温度や電圧などのパラメータを定義することもできます。[バッテリーセレクション]は、バッテリータイプを選択または定義するために使用します。バッテリーソースはオプションであり、定義されている場合は選択されたシーケンスステップでのみ使用でき、独立して動作し外部電源に接続されたデバイスをシミュレートします。情報とヘルプ セクションには、ユーザーに役立つメモが含まれています。

- シーケンスは順序付けられたステップをセット

既存のステップを追加または複製し、新しいステップを作成

既存のシーケンスをロードして適合

異なるMCUを使用してシーケンスを比較

提案された電源ステップの遷移が有効かどうかを自動的に確認

Step	Mode	V _{DD}	Range/Scale	Memory	CPU/Bus Fr.	Clock Config	Peripherals	Step Current	Duration
1	RUN	1.8	Range1-High	FLASH	16 MHz	HSE BYP	ADC1fs_10_ksp...	2.55 mA	1 ms
2	STANDBY	1.8	NoRange	FLASH	16 MHz	HSI		770 nA	1 ms
3	LOWPOWER_RUN	1.8	NoRange	FLASH	2 MHz	HSI Regulator_LP	GPIOA GPIOB GPIO C...	496.73 µA	1 ms
4	LOWPOWER_SLEEP	1.8	NoRange	Flash-Power...	1 MHz	HSI Regulator_LP	IOPORT_Bus RTC RTC...	243.3 µA	2 ms
5	VBAT	3.0	NoRange	FLASH	16 MHz	HSI Regulator_LP		2 nA	1 ms
6	RUN	1.8	Range2-Medium	SRAM1 Flash...	4 MHz	HSE BYP	GPIOA GPIOB GPIO C...	559.26 µA	1300 µs
7	SLEEP	1.8	Range2-Medium	FLASH	8 MHz	HSE BYP	GPIOA GPIOB GPIO C...	489.02 µA	2 ms
8	STOP0	3.0	NoRange	Flash-Power...	16 MHz	HSI_KERON		308 µA	500 µs

「シーケンステーブル」では、異なる期間と構成を持つ一連のステップを設定します。その長さは事実上無制限です。

シーケンスは、ロード、変更、再利用できます。

個々のステップは、ユーザーインターフェイスを使用してシーケンス内で複製および再配置できます。

有効にすると、すべての状態遷移が、頻度または電力範囲で不正にジャンプすることを防ぐために、基本的な有効性ルールに対してチェックされます。問題のあるステップは、シーケンステーブルで即座に強調表示されます。

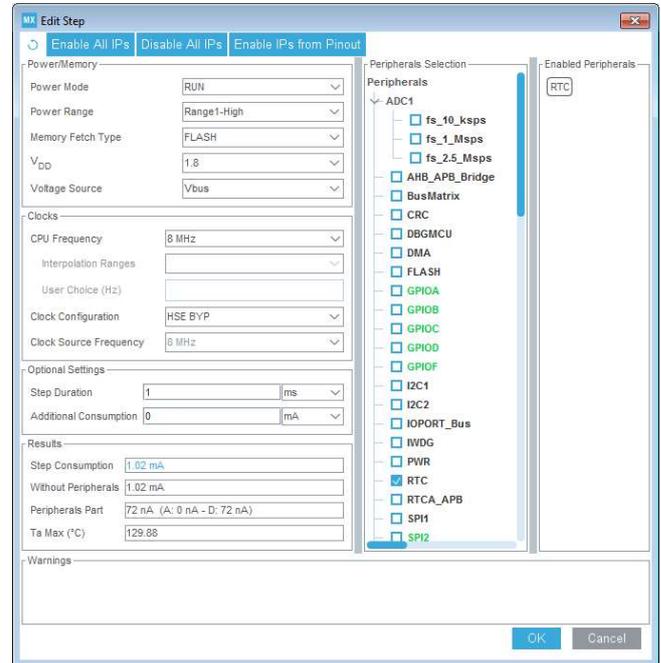
「ログを表示」ボタンをクリックすると、詳細な説明が表示されます。

「比較」機能は、保存されたシーケンスと現在のシナリオでのパワーとパフォーマンスの比較を表示します。異なるMCUsを含む異なる構成は、相互に対して評価することができます。

消費電力設定ステップ定義 1/2

30

1. 電源モードの選択により、どのペリフェラルが使用可能かがわかる
2. レギュレータの設定は、パフォーマンスと消費のバランスが必要
3. コード実行元のメモリ、およびプリフェッチとバスオプションを選択
4. V_{DD} – いくつかの設定が利用可能
5. このオプションはバッテリー寿命の計算のために用意されている



電源ステップは、このダイアログウィンドウで追加または編集できます。トランジションチェッカーが有効になっている場合、新しいステップに許可された値が事前設定されます。

電源ステップはいくつかの特性によって決定され、電源モードが最も重要なパラメータです。各電源モードの可用性と特性については、リファレンスマニュアルまたはデータシートに記載されています。電源モードの選択は、他の設定、インタフェイス、および電源/パフォーマンスバランスの可用性に最も大きな影響を与えます。

電圧レギュレータは、コア電圧を設定します。低電圧ではシステムクロック周波数が制限されますが、消費電力が大幅に削減されることが期待できます。詳細については、データシートを参照してください。命令が取り出されるアドレスと関連する設定は、電力消費と使用可能なクロック速度にも影響を与えます。

電力消費を計算する電源電圧。実際の電圧が使用できない場合は、最も近い値を使用します。

最後のオプションは、デバイスがバッテリードレインモデルでUSBに接続されている場合など、ケースを除外するために存在します。

電源モードの詳細については、システム電源制御モジュールのトレーニングプレゼンテーションを参照してください。

• クロック

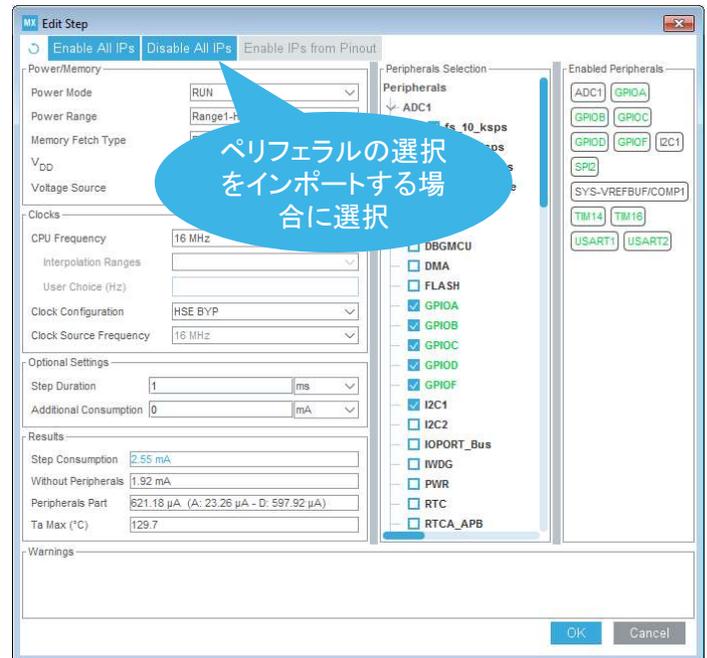
- 動作周波数の選択は、電力消費範囲によって制限される
- 使用可能なクロック構成は、利用可能なデータおよびその他の設定によって異なる

• ペリフェラル

- ペリフェラルに対するクロック・ゲーティングを選択
- ピンアウトタブから選択をインポート

• オプション設定

- 追加の消費電力は、ピンに接続される推定負荷で表示



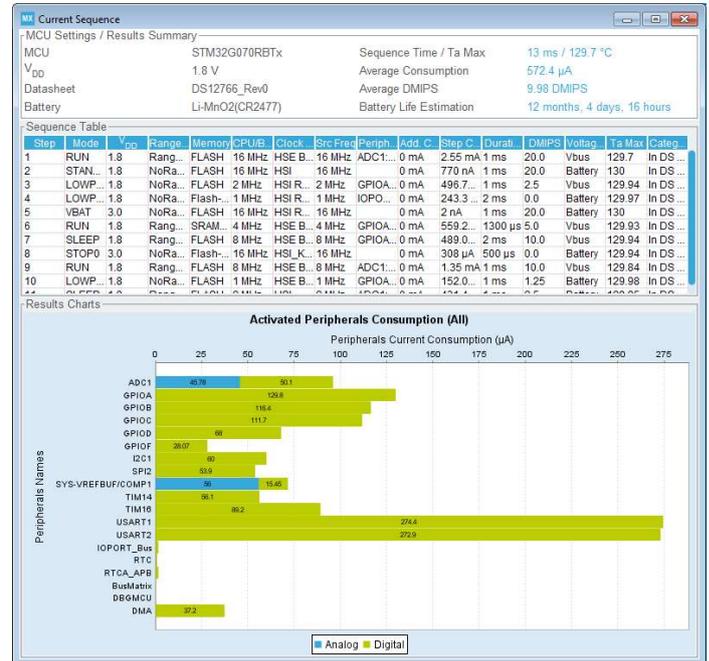
クロックの設定は、電源とメモリの両方の設定と、使用可能な測定データによって制限される場合があります。オプションの完全なリストについては、ドキュメントを参照してください。

使用されていないペリフェラルのクロックを無効にすると、確実に消費電力を節約できます。ステップの持続時間中に電源が供給できるペリフェラルを確認できます。「インポート」ボタンをクリックすると、生成されたコードで初期化されるすべてのペリフェラルが選択されます。

最後に、ステップ期間と追加の消費設定を定義します。追加の消費は、LED、ボタン、通信インタフェースなど、ピンに接続されているさまざまな負荷によって表されます。

消費プロファイルのシーケンス表示

- プレゼンテーション目的のため外部表示にグラフを切り離すことが可能
- 複数の異なるビューを選択可能
 - 現在と時間のプロット
 - 円グラフ
 - ペリフェラルの消費



消費電力計算機は、強力なプレゼンテーションツールを備えています。[Ext. Display]ボタンをクリックすると、レポートが別のウィンドウに表示されます。

現在の消費見積もりをグラフィカル形式でプロットするには、さまざまな方法があります。デフォルトの方法は、電源ステップのシーケンスと時間経過に対する消費に基づいています。

または、さまざまなモードで消費されるエネルギーの割合をグラフ化することもできます。円グラフには、各モードのシェアが表示されるか、または実行モードと低電力モードのみが表示されるように分割できます。

また、ペリフェラルの消費電力を分割し、その電力要件をグラフにプロットすることも可能です。デジタルペリフェラルのみ、アナログペリフェラルのみ、または両方を含む混合ビューをプロットできます。

- オプションの手順として、PDFレポートを生成
- PDFレポートはPCCなしでも利用可能
- 保存されたプロジェクトには、次の項目が含まれる
 - Project.ioc
 - Project.pcs
 - Project.pdf
 - Project.txt
 - Project.jpg
 - +サポートされている開発環境用に生成されたプロジェクト

6. Power Consumption Calculator report

6.1. Microcontroller Selection

Series	STM32G0
Line	STM32G0x0 Value line
MCU	STM32G070RBTx
Datasheet	DS12766 Rev0

6.2. Parameter Selection

Temperature	25
Vdd	1.8

6.3. Battery Selection

Battery	Li-MnO2(CR2477)
Capacity	850.0 mAh
Self Discharge	0.12 %/month
Nominal Voltage	3.0 V
Max Cont Current	2.0 mA
Max Pulse Current	10.0 mA
Cells in series	1
Cells in parallel	1



life.augmented

拡張子が .ioc のファイルには、静的な初期化設定が含まれています。電源シーケンスは.pcs拡張子を使用して保存されます。PDF レポートが生成され、簡略化されたテキストと、ピンアウトされた個別の JPG イメージ ファイルが生成されます。

- 詳細は以下の資料を参照
 - UM1718 –ユーザーマニュアル
 - DB2163 –製品仕様
 - TN0072 –製品テクニカル ノート
 - RN0094 –製品リリースノート

- STウェブサイトからツールをダウンロード www.stmcu.jp



life.augmented

STM32CubeMXコード生成ツールの使用に関する詳細について、
またこのスライドに記載されているドキュメントはwww.stmcu.jp
からダウンロードできます。
ありがとうございました。