



## STM32L4 – STM32CubeMX

STM32CubeMX開発ツール

Revision 1

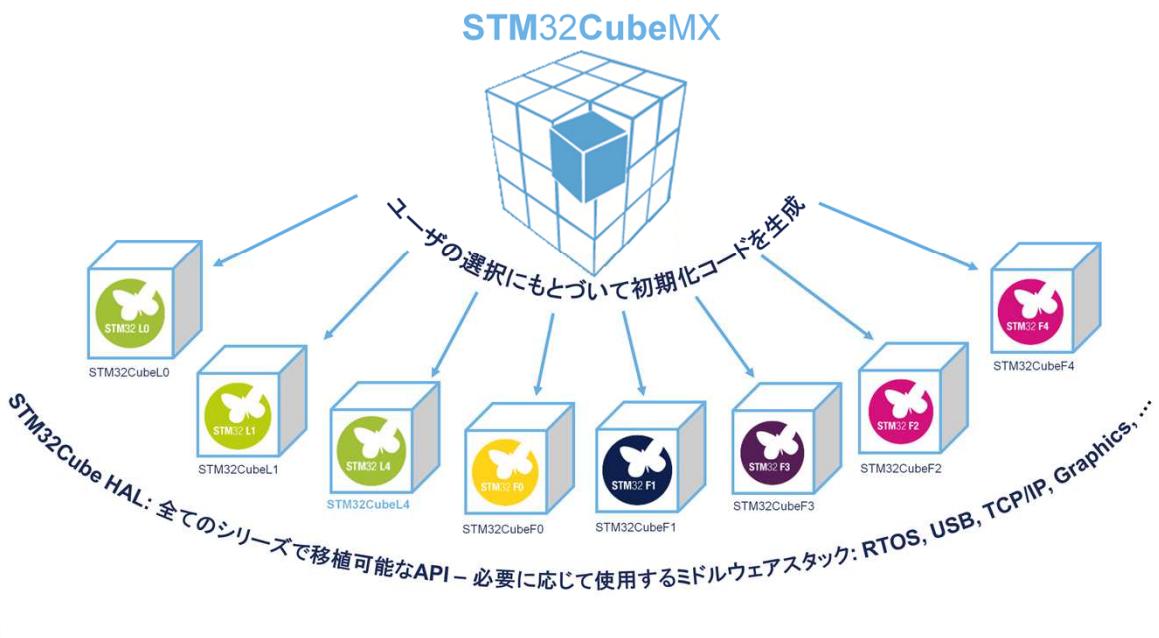


Jan - 2016

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。このプレゼンテーションではSTM32CubeMXの主機能を網羅します。その主機能はSTM32ファミリ向けの初期化コードを生成に使用されます。

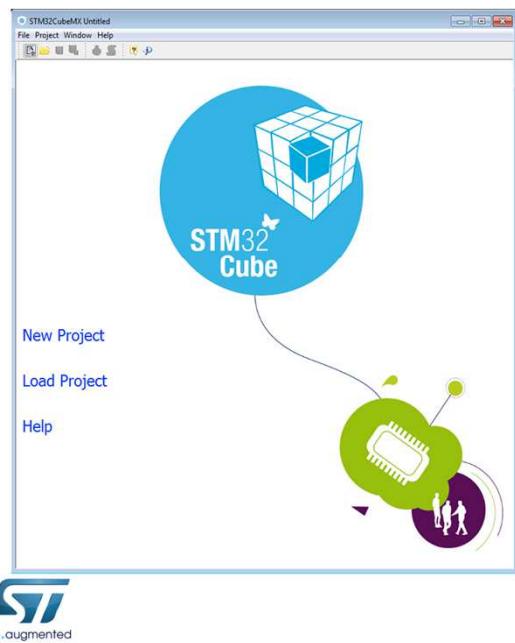
# STM32CubeMXの紹介

2



このプレゼンテーションはSTM32L4シリーズを取り上げますが、STM32CubeMXはSTM32ファミリ全体で共通のプラットフォームです。





- 最適なマイコンを選択し下記を設定
  - ピンアウト
  - クロック及びオシレータ
  - ペリフェラル
  - 低消費電力モード

### 使用時の利点

- 使用目的に合致したマイコンを選べます。
- 設計段階で有益なシミュレーションを実行できます。
- 初期開発スピードをより速くします。

STM32CubeMXアプリケーションはGUIでファームウェアプロジェクトの初期設定をガイドし、STM32マイコンを使用する開発者の作業を支援します。

そのGUIはピン配置、クロックツリー、内蔵ペリフェラルの設定、及び、生成されるファームウェアプロジェクトの電力消費のシミュレートする手段を提供します。

また、そのGUIはSTM32マイコン製品ポートフォリオ由来の豊富なデータライブラリを使用します。

このGUIアプリケーションは、開発者が機能及び電力の観点で最適なマイコンを選択する作業を助け、開発初期段階の作業を簡単なものすることを目的としています。

- ・マイコン選択
  - ・製品ファミリ、パッケージ、及びペリフェラルもしくはメモリサイズによる自動絞込み。
- ・ピン配置設定
  - ・使用するペリフェラルの選択、GPIOの割り当て、ピンに対するオルタネート機能の割り当て。
- ・クロックツリー初期化
  - ・オシレータの選択、PLL及びクロック分周設定。
- ・ペリフェラル及びミドルウェアパラメータ設定
- ・電力消費計算機
- ・ソースコード生成
  - ・変更したくないユーザコードを保持したまま再生成可能。



適切なマイコンを選択し、必要なペリフェラルを選択し、ピン設定を割り当てる自然なワークフローを実現するようにユーザ・インターフェースが構成されています。

電力消費計算機は効率の良いシステム設計を目的としています。最後にプロジェクトの初期化コードを生成することができ、また変更したくないユーザコードを保持したまま再生成も可能です。

# 使用時必要条件及び設定

5

- STM32CubeMXにはJava REが必要です。
  - 追加の要件についてはSTM32CubeMXのリリースノートをご確認ください。
- インストール後、アップデータの設定のためAlt+Sを入力します。更新対象にはGUIツールだけでなくCubeファームウェアライブラリも含まれます。
- ソフトウェアライブラリの配置場所を選択します。



STのウェブサイトからSTM32CubeMXの無償インストーラをダウンロードし、インストールします。

そして、メニュー設定でユーザ設定を調整します。

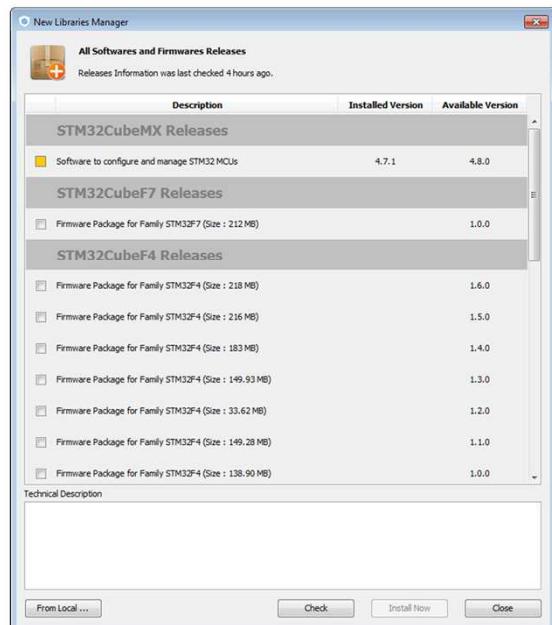
- アップデータ及びライブラリダウンロードのメニュー(Alt+S)。
- コード生成及び開発ツールチェインとの結合メニュー(Alt+P)。

この設定を済ませると、新しいプロジェクトが生成可能になります。

## アップデータ

6

- ・アップデータはヘルプメニューから選択可能です。
- ・ツールアップデータはツールの更新及び関連するSTM32Cubeライブラリの新リリースを検出可能です。
- ・新ライブラリパッケージのダウンロードにはダウンロードマネージャを使用します。



インターネット接続が適切に設定されていれば、プロジェクトワークスペースの生成に使用されるGUIツール及びコードライブラリが更新されます。

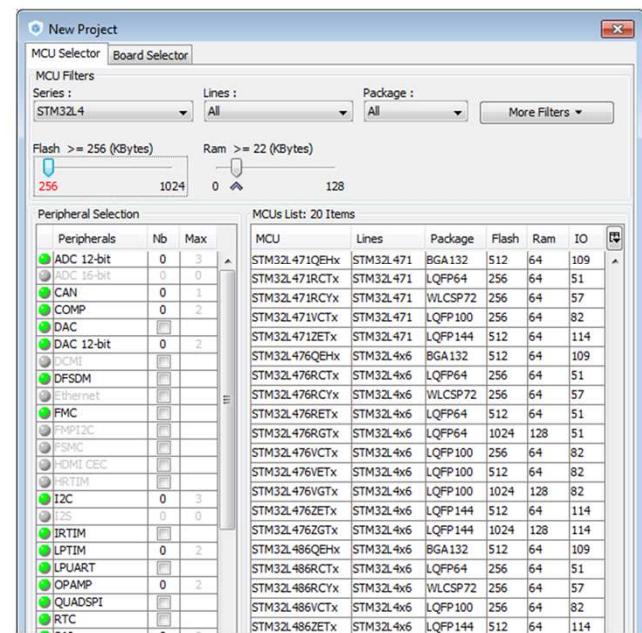
「新ライブラリのインストール」オプション(Alt+U)を使用して追加のSTM32Cubeライブラリをダウンロードするか、もしくはソフト開発時のバージョン維持目的の場合は古いバージョンを取得します。

しかしながら、STM32CubeMXツールはリリース済のライブラリ全バージョンでテストされておらず、古いSTM32CubeMXツールでは新しくリリースされたライブラリが動作しない可能性があります。

# マイコン選択

7

- データベースには数百の製品が記録されています。
- 名前によるマイコンの選択 ...
  - シリーズ・ラインによって手早く表示されます
- ... もしくは用途による選択
  - パッケージ
  - RAMサイズ
  - 不揮発メモリ
  - 内蔵ペリフェラル
  - インターフェースの種類、数



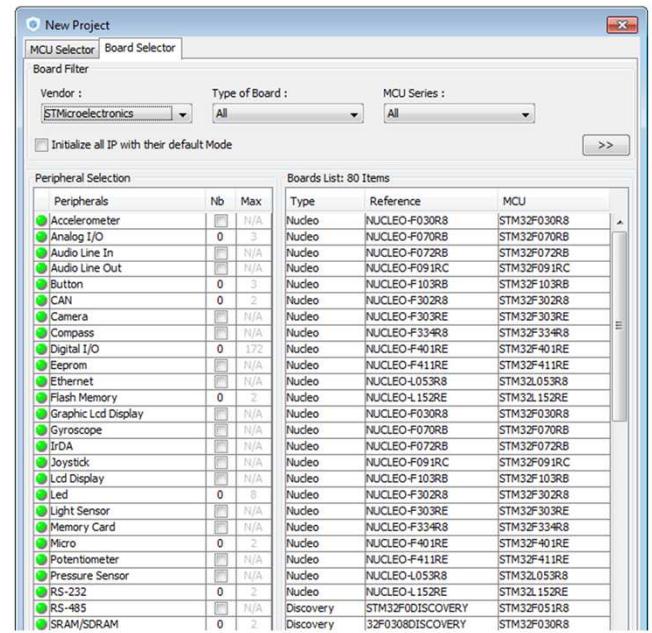
“New Project”を選択するとマイコン選択ウィンドウが表示されます。ユーザがどのマイコンを使うか事前にわかっていれば、すぐに見つけることができます。

どのマイコンを使うかまだ決まっていなければ特定の要求にもとづいて使用可能な製品を選別可能です。

## マイコン選択(続)

8

- もう一つのタブはSTM32マイコンを搭載したST製既存評価ボードへのショートカットです。
- 既存ボードには既にピン配置が割り当てられており、そのボードごとの機能や接続を使用します。
- 他のボード設定は網羅されていません。



“Board Selector”タブで既存のSTM32評価ボードに対する設定が可能です。例えばSTM32L476-EVALボードが選択された場合、LCD、ボタン、オーディオ、及び、通信インターフェース用のI/O群がロードされます。ただし、そのボード上のいくつかの通信インターフェースはジャンパもしくは半田ブリッジを調整した後に使えるようになるため、STM32CubeMXツールでは設定済になっていません。

## ピン配置

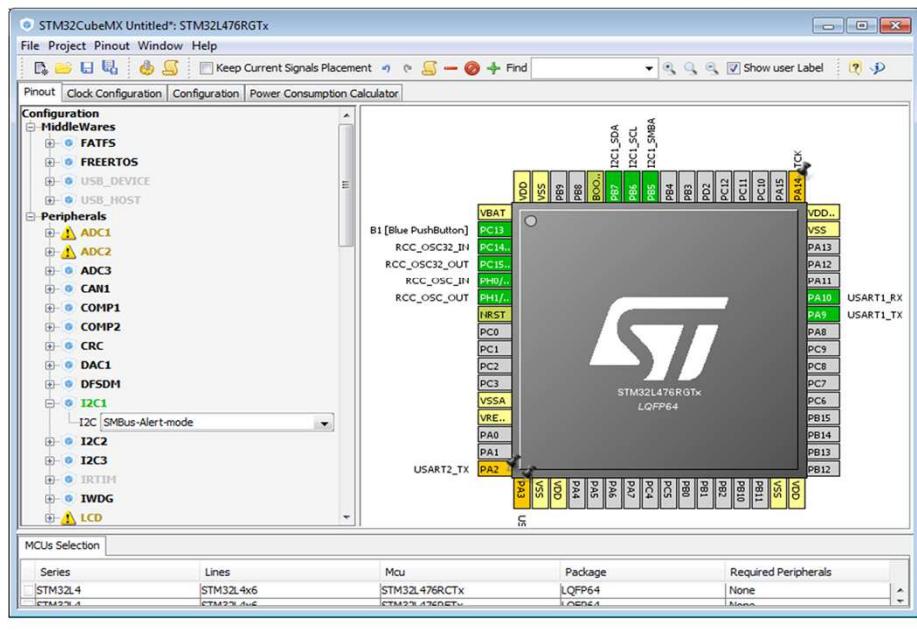
9

- ・下記によるピン配置

- ・ペリフェラルツリー
  - ・手動

### ・自動信号リマッピング

- ・ペリフェラル、ミドルウェア(FatFS、USB…)  
固有の依存性管理



次のステップは使用するペリフェラルを選択し、選択したペリフェラルの入出力にピンを割り当てることです。

ペリフェラルに依存しない独立した汎用GPIOの設定も可能です。信号はデフォルトピンに割り当てられていますが、Ctrl+クリックで表示される代替ピンに移すこともできます。

例えばI2C1が有効にされるとツールは自動でデフォルトピンに割り当てます。

STM32CubeMXツールは管理しているペリフェラルとソフトウェア部品の間に存在するほとんどの組み合わせを自動で考慮します。

## ピン配置(続)

10

ペリフェラルのオルタネートピンが全て別の用途に使用されている場合、赤で表示されます。

ピンアイコンで信号ピンを配置します。

ピンをクリックするとオルタネート機能が表示されます。

オレンジはペリフェラルが割り当てられているだけで使用可能な状態になっていないことを示します。



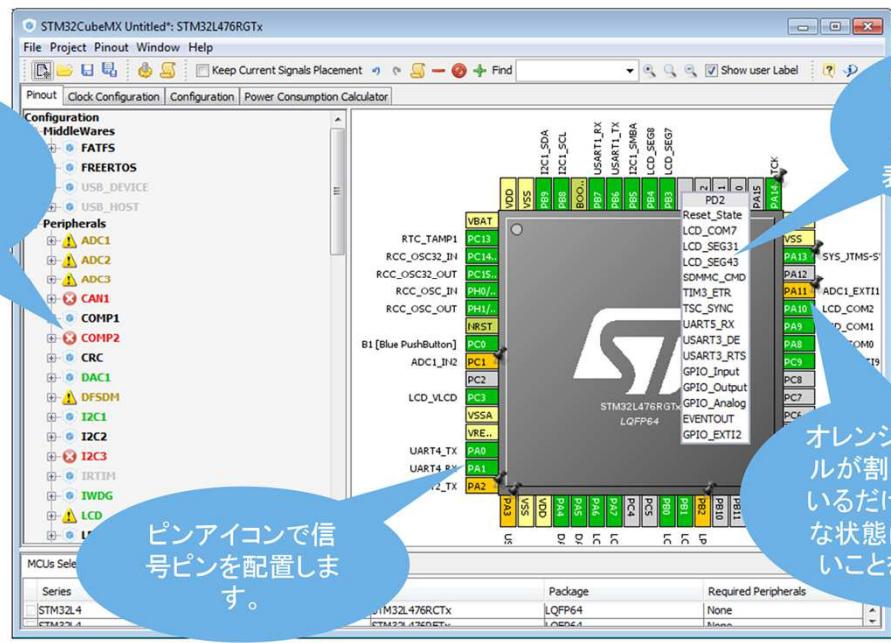
より多くのピンがペリフェラルのオルタネート機能向けに予約されているため、別のペリフェラル向けに選べるピンはより少なくなります。

そのピン割り当て上の制約は他のペリフェラルノードのアイコンの変化で表示されます。

ピンの上で左クリックするとオルタネート機能が表示されます。

ピンの上で右クリックするとピン割り当てを選択できます。

特定のペリフェラルを有効にしないでピン配置を選択するか、もしくはピン配置に問題がある場合は、そのピンが緑ではなくオレンジ表示になります。



- ペリフェラル・モードには3つの状態があります。
  - 灰色: そのペリフェラル・モードは使用できません。他のモードにする必要があります。
  - 黄色: そのペリフェラル・モードは制限付で使用できます。
  - 赤色: そのペリフェラル・モードに必要な信号がピンに割り当てられない状態です。
- ピン割り当てビューから信号を直接、設定・移動できます。
  - ピンをクリックして割り当て可能な信号リストを表示し、そのうちのひとつを選びます。
  - ある信号を割り当て可能なピンを表示するには、その信号をCtrl+クリックして別の場所にドラッグします。
  - 未使用のピンは無視して、コード生成時にアナログモードにすれば消費電力を抑制できます。



ペリフェラルモードには3つの状態があります。

- 灰色: そのペリフェラルモードは使用できません。他のモードにする必要があります。灰色のモードの上にマウスポインターを置くと使用できない理由を見れます。例として、クロックソースを無効にする必要ありことや、他のペリフェラルと依存関係があることなどがあげられます。
- 黄色: そのペリフェラルモードは制限付で使用できます。複数のオプションが競合しているため制限付になっています。例えば全ての選択可能なUSARTクロックピンが他の用途に使用されているとUSARTを同期モードで設定できなくなります。
- 赤色: そのペリフェラルモードに必要な信号がピンに割り当てられない状態です。例えば主要な信号の割り当てに必要なピン全てが他のペリフェラルに使用されている場合が該当します。

ピン配置ビューから信号を直接、設定・移動できます。

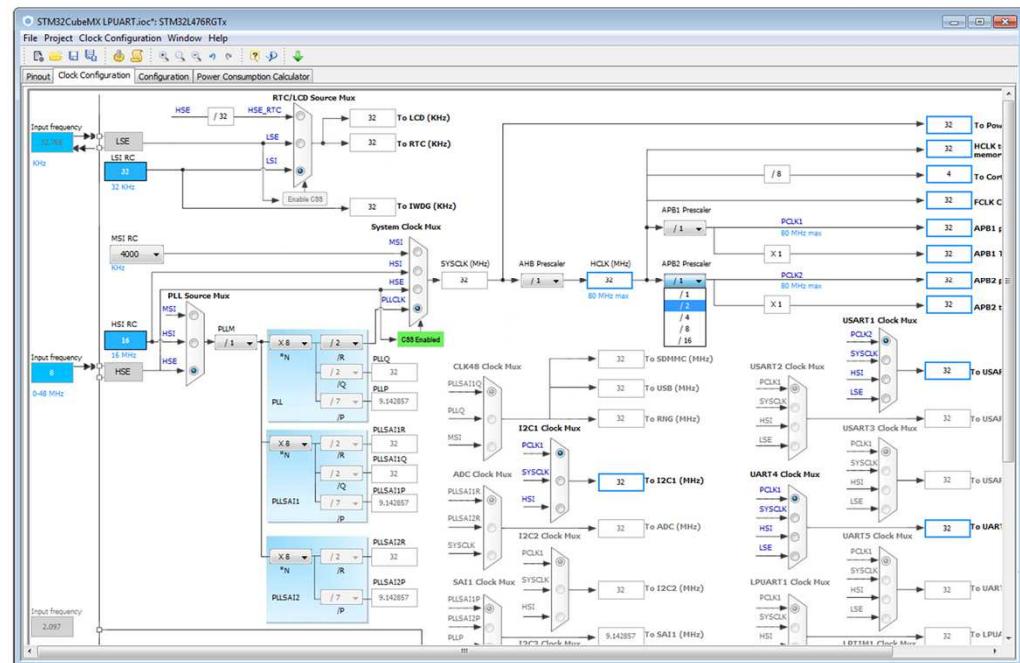
- ピンをクリックして割り当て可能な信号リストを表示し、そのうちのひとつを選びます。この操作はペリフェラルに割り当てられていないGPIOに対して動作します。

- ある信号を割り当て可能なピンを表示するには、その信号を Ctrl+クリックして別の場所にドラッグします。Ctrlキーの押下したままドラッグ & ドロップで信号を別のピンに移動できます。
- 未使用のピンを手動でアナログモードに設定する必要はありません。未使用のピンを半自動でアナログモードに設定できます。

# クロック設定

12

- ・全てのクロック設定値の俯瞰
- ・有効および無効なクロックパスを識別
- ・クロック制約と機能の管理

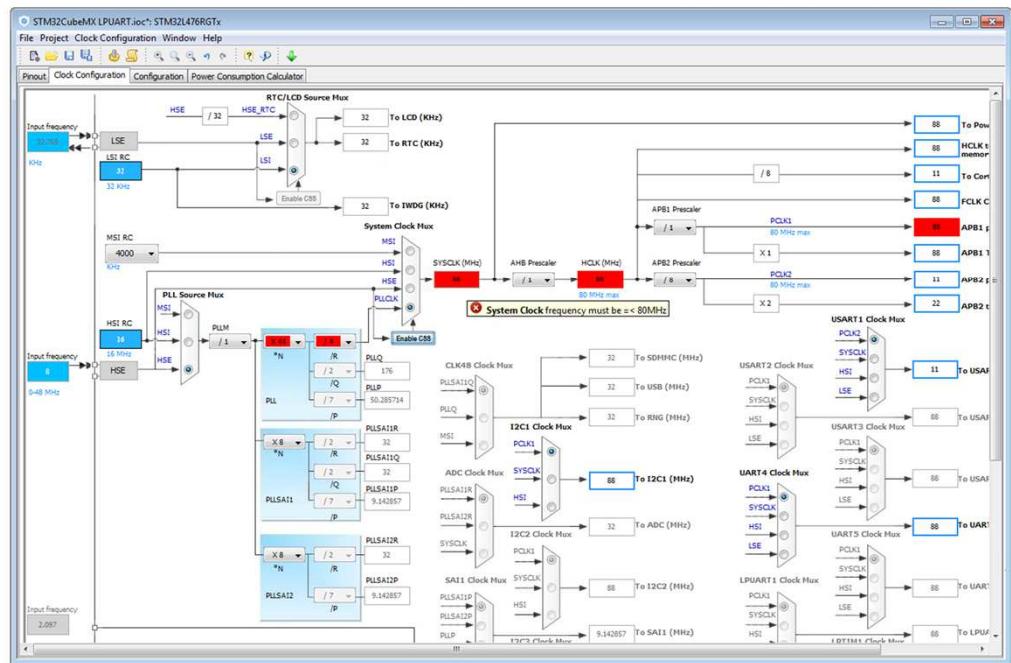


クロック設定タブで、全てのクロックソース、分周器、遡倍記を網羅したクロックパスの配線図、および、実際のクロックスピードが見れます。有効な信号は青色でハイライトされます。  
ドロップダウンメニューと制御ボタンでクロック設定を調整します。

## クロック設定(続)

13

- エラーのハイライト  
赤で表示されます。
  - 青色の枠に数値を  
入力し、ツールに  
分周器、遅倍記の  
調整を促します。
  - 入力数値を固定し  
てツールによる調  
整を抑制します。



範囲外の数値を設定すると、問題の発生を知らせるため表示が赤になります。

青色の枠に必要なクロックスピードを入力すると、そのスピードになるようにソフトウェアが分周器、遅倍記を再設定します。青色の枠のクロック値を右クリックすると固定され、変更から保護します。

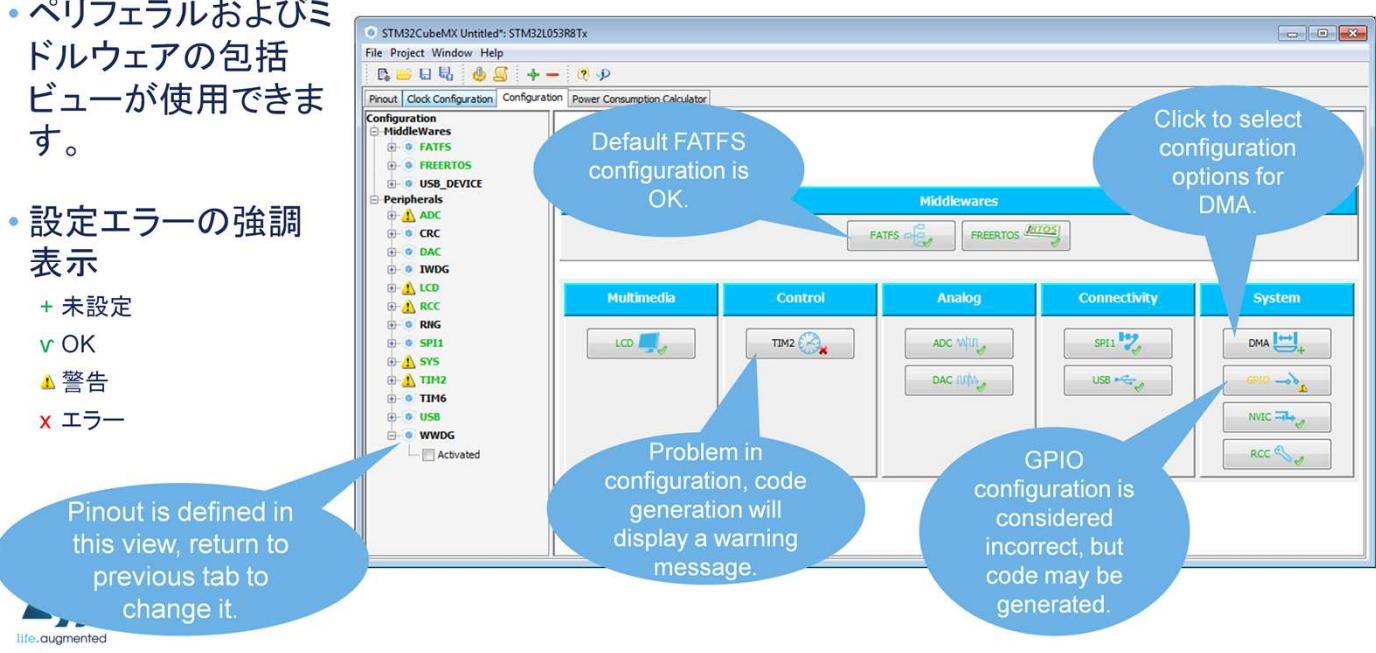
# ペリフェラル、ミドルウェア設定

14

- ペリフェラルおよびミドルウェアの包括ビューが使用できます。

- 設定エラーの強調表示

- + 未設定
- ✓ OK
- ⚠ 警告
- ✗ エラー



メインウィンドウの設定タブでSTM32CubeMXの設定支援対象となる全てのハードウェアおよびソフトウェア部品を俯瞰できます。設定オプションにアクセスするボタンは設定可能状態を示す小さいアイコンとともに表示されます。

デフォルトの状態は設定できません。ボタンをクリックするとペリフェラルもしくはミドルウェアがそれらの設定オプションとともに表示されます。

適切に設定した後でも更に変更できます。

設定をまちがえると警告サインが通知されます。その状態でコードを生成してしまうとペリフェラルが動作しません。

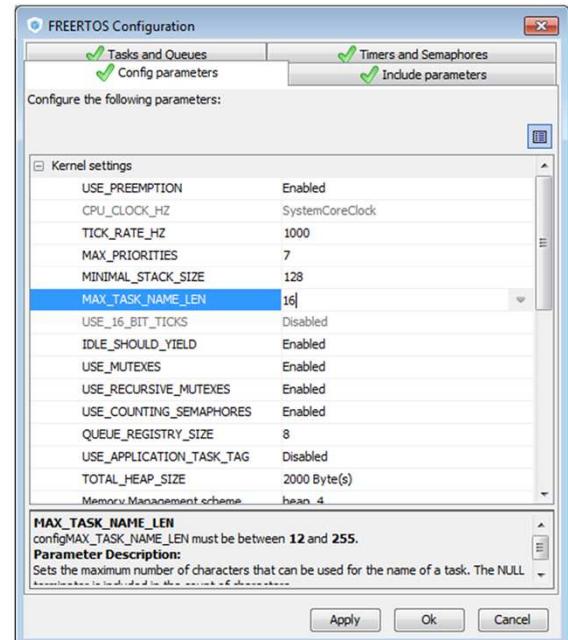
ひどいまちがいは赤色のXで表示されますので設定を調整する必要があります。

設定するペリフェラルやミドルウェア部品を追加するにはピン配置タブに戻ります。

## ミドルウェア設定

15

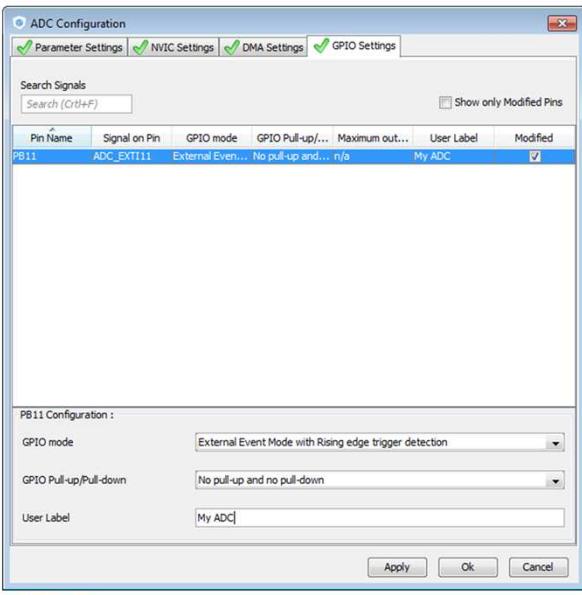
- ・サポートしている各ソフトウェア部品固有のオプションを提供します。
- ・全ての設定は論理グループごとに分けられています。
- ・クイックリファレンスとして説明や制約条件が使えます。



各ミドルウェア部品のオプションはそれぞれ異なりますが同じような様式で表示され、初期化オプションを簡単に把握でき、その他有益な情報を提供します。

# ペリフェラル設定

16



- 設定可能な初期化パラメータがオプションと簡単な説明とともに提供されます。
- 割込みはペリフェラルに割り当てることがあります。
- 使用可能なDMAも紐づけられています。
- ペリフェラルで使うGPIO入出力も紐づけられています。

ペリフェラルを設定する際、ダイアログウィンドウは基本パラメータ、依存関係、および制約条件を表示します。パラメータが複数ある場合はドロップダウンメニューが使用されます。

割込み優先度は”NVIC”タブで設定できます。ペリフェラルウィンドウでは割込みの有効・無効だけ設定できます。

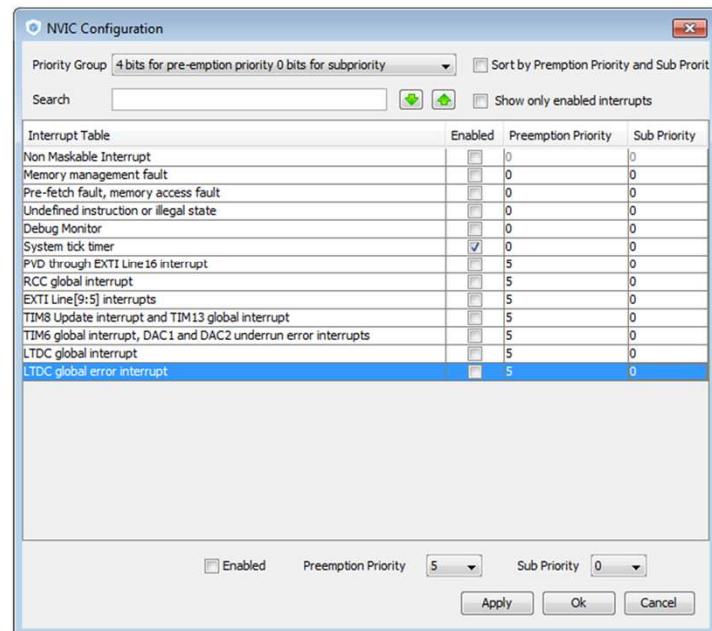
DMA設定タブではDMAリクエストの初期化に関連するパラメータを設定でき、転送開始アドレス等の実行時パラメータは設定できません。

GPIO設定タブはGPIOパラメータや機能の設定に使います。

## NVIC設定パネル

17

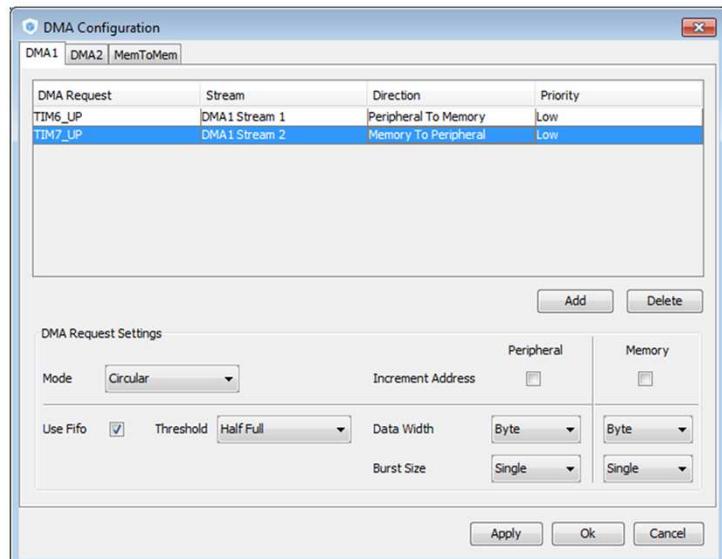
- ・全ての割込みに対してひとつの制御パネル
- ・優先度およびサブ優先度を管理
- ・一覧表での割込みの検索、絞込み、並べ替え用インターフェース



STM32CubeMXのもうひとつの特徴は、使用可能な割込みおよび有効な割込みを優先度情報つきで一目で把握できることです。このウィンドウは選択されたペリフェラルの割込みを有効にし、割込み優先度を設定する用途に使います。

# DMA設定パネル

18



- メモリからメモリへの転送を含めた全てのDMAリクエストを管理
- 転送方法、優先度、およびその他の設定を調整



対応するDMAチャンネルタブを選択して"Add"をクリックすると特定のペリフェラル向けのDMAリクエストが追加されます。また、全ての設定オプションを検証します。この設定パネルでDMAチャンネルを設定できますがDMA転送に関わる全ての条件を設定できるわけではない点にご注意ください。その残りの設定はアプリケーションコードで記述し、実行することになります。

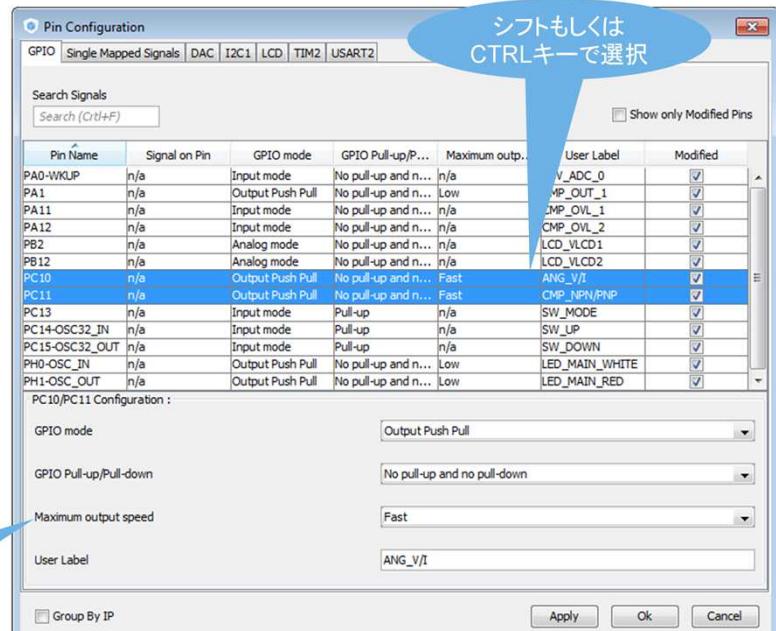
# GPIO設定パネル

19

- このアプリケーションは、ほとんどのGPIOパラメータに的確なデフォルト値を設定しようとします。
- デフォルト値には低速スピード、プルアップ無等の控えめなものが選択されます。
- 複数のピンを選択して同時に同じ設定にできます。



スピードが正しいか確認



ピン設定ウィンドウのGPIOタブで各ピンの初期化を簡単に設定できます。

各ピンが一覧表形式で表示されユーザー ラベル付でピン設定を一目で把握できます。

ドロップダウンメニューと使って並べ替え、検索、設定調整を選択したピンに実行できます。

ツールが割り当てたデフォルト値は安全です。しかしあるペリフェラル設定に対しては動作しない場合があります。

ツールが選んだGPIOスピードがペリフェラルの通信スピードに対して充分か、内部プルアップが必要なピンに選んであるかについてはユーザーが確認する必要があります。

より手早くピンを設定したい場合は、各ピンを個別に設定せずにピンをグループごとに選びます。特定のペリフェラル専用のピングループを得るにはペリフェラルタブを使用します。

初期化時の設定は実行時に変更可能ですが、実行時の変更はSTM32CubeMXツールの適用範囲外となります。

## コード生成

20

- C言語による初期化コードを生成します。
- サポートする開発ツール向けにプロジェクトファイルを生成します。
- 特定の場所にユーザーコードを追加でき、コード再生成時にも保持されます。
- コード再生成時であっても最新バージョンのライブラリを使用するか使用中のバージョンを維持するか選択可能です。

```
main.c
22  /*
23  */
24  /* Includes
25  *  stm32f4xx_hal.h
26  *  cmsis_os.h
27  *  lvip.h
28  *  usb_device.h
29
30  /* Define structures */
31  ADC_HandleTypeDef hadc1;
32
33
34  /* USER CODE BEGIN 0 */
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 interface
51  *  HAL_Init();
52  *  Configure the system clock */
53 }
```



全ての入出力、ペリフェラルの設定が完了すればコードを生成できる段階になります。

まず最初にメインウィンドウのプロジェクトメニューで各設定にチェックを入れます。

サポートしている開発ツールKeil、IAR、Atollic、およびAC6からひとつ選択します。

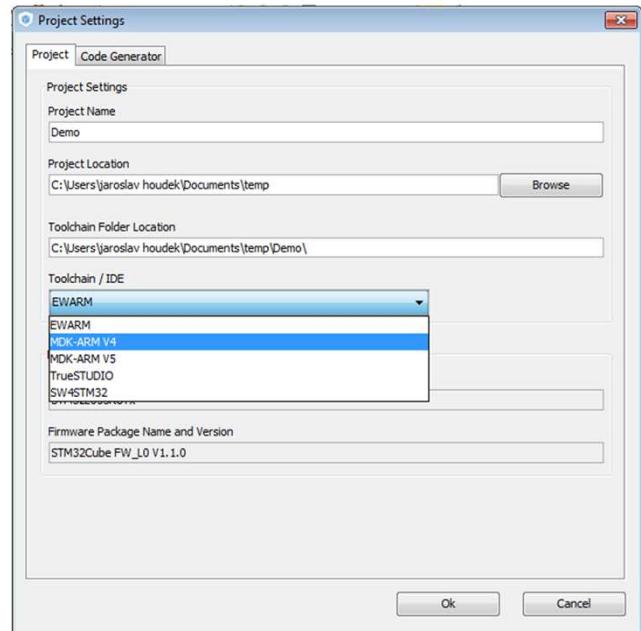
ユーザーコードを”USER CODE”コメントブロック内に収めておけば初期化設定を変更しコード再生成した場合でも消されることなく保持されます。

次のスライドでこのオプションを有効にする方法を説明します。

# コード生成時プロジェクト設定

21

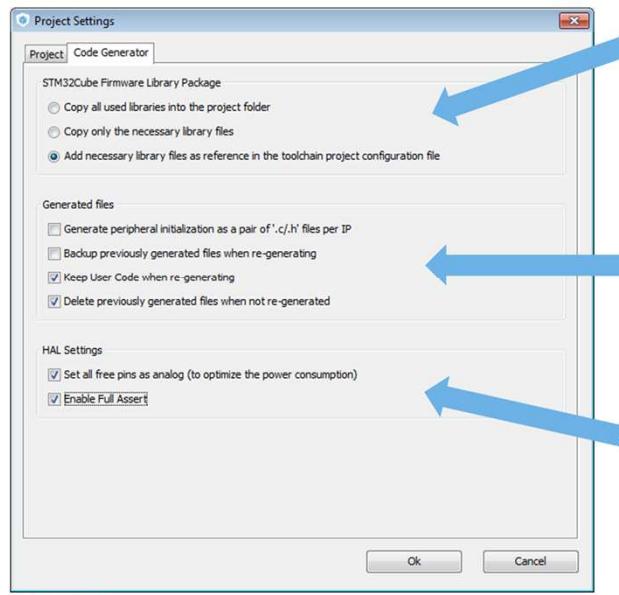
- ・保存する際のプロジェクト名
- ・プロジェクト配置場所の閲覧
- ・使用するツールチェインの選択
- ・マイコンタイプ、ライブラリバージョンのレビュー



このウィンドウは(Save as...)でプロジェクトを保存する際に使用できます。ツールチェインフォルダはツールチェインそのものがどこにあるかではなく、そのツールチェイン向けのワークスペースがどこに配置されるかを示しています。このダイアログウィンドウの簡易版をAlt+Pショートカットで使用でき、プロジェクト設定を表示できます。

# コード生成オプション

22



## • ライブラリパッケージ

- これから生成されるプロジェクトフォルダにライブラリ全体もしくは必要なものだけコピーされます。
- もしくはライブラリをもともとある場所に維持して全てのプロジェクトから参照します。

## • ファイル生成

- 各ペリフェラルを別々のファイルもしくは共通のソースファイルで初期化します。
- 古いファイルを取り扱うためのオプション。
- ユーザーコードをそのまま維持するオプションはここで選択します。

## • HAL設定

- 未使用ピンをアナログモードに設定することで消費電力を抑えることができますが、ピン配置タブで明示的にSWD/JTAGを選択する場合は注意が必要です。
- デバッグに有用なフルアサート。

STM32Cube HALライブラリは様々な方法でプロジェクトと連携します。プロジェクトをコンパクトなパッケージとして移植する必要がある場合や、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コピーオプションを選びます。ライブラリをもともとある場所で維持して使うと複数のプロジェクトで最新版ライブラリを共有しやすくなります。

全てのペリフェラルの初期化コードをstm32fxx\_hal\_msp.cに集約したり、ペリフェラル毎にひとつの初期化コードファイルを生成したり、どちらもできます。

古いファイルのバックアップもしくは削除のどちらを選ぶかはワークフローの好みによります。これらのオプションは再生成機能で使用します。また、"keep user code when re-generating"オプションもここで選択できます。

"Set all free pins as analog"設定は消費電力の低減に役立ちますが、もしSWD/JTAGインターフェースをピン配置タブで選択していない場合、このアナログ端子設定オプションはデバッグインターフェースを無効にしてしまいます。

"Full assert"はHAL関数に渡される引数のチェックを有効にし、

ユーザーコード上のバグの検出に役立ちます。

# 注意点と免責事項

23

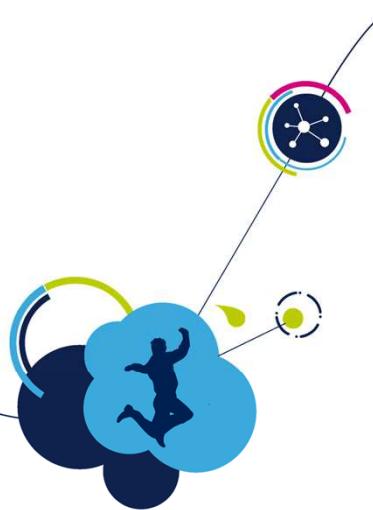
- STM32CubeMXツールはSTM32ファミリ全体に対するユニバーサルデザインという位置づけであるため特定の製品のみが持つ特定の機能にフォーカスしていません。
- STM32CubeMX GUIツールはリファレンスマニュアルやデータシートを置き換えたものはありません。
  - 詳細な情報は必ず文書を参照してください。
  - 重要な機能はマイコン製品もしくはHALドライバで使用可能です。
- GUIツールはプロジェクトの開始および設定初期化の面で有用ですが、設定は実行時に動的に変化します。(例:GPIO、NVIC優先度、クロック設定等)



TSTM32CubeMXユーザーインターフェースは有用なツールですが、機能上の制限があります。全てのSTM32マイコンをユニバーサルアシストするものであるため、STM32ポートフォリオを俯瞰するのに有用ですが、各マイコン製品の全ての細部を追及するものではありません。

迷った場合はリファレンスマニュアルやデータシートを参照しより詳細かつ正確な情報を調べてください。更に学習する場合はアプリケーションノートや参考使用例を使用してください。STM32CubeMXで動作するプロトタイプを手早く作成してアプリケーション開発に着手し、(同じアプリケーションで異なるクロック、GPIO設定が必要になる場合等)必要に応じてコードを編集していく、というのが想定使用例になります。

STM32CubeMXジェネレータが区割りしたユーザーエリア内でコードを記述していて、GPIOピンの設定を追加したり、クロックを変更したり、NVIC優先度を変更したくなった場合は、STM32CubeMXの初期設定に戻ればそれらの変更が可能です。



## 電力消費計算機プラグイン



組込みアプリケーション開発の際、低消費電力は重要な設計目標となります。とはいえ、データシートから電力消費情報を抽出するのは時間がかかり退屈な作業です。電力消費計算機はデータシートから電力消費情報を抽出する作業をユーザーインターフェースツールで置き換えて設定シナリオから有益な見積もりを生成します。

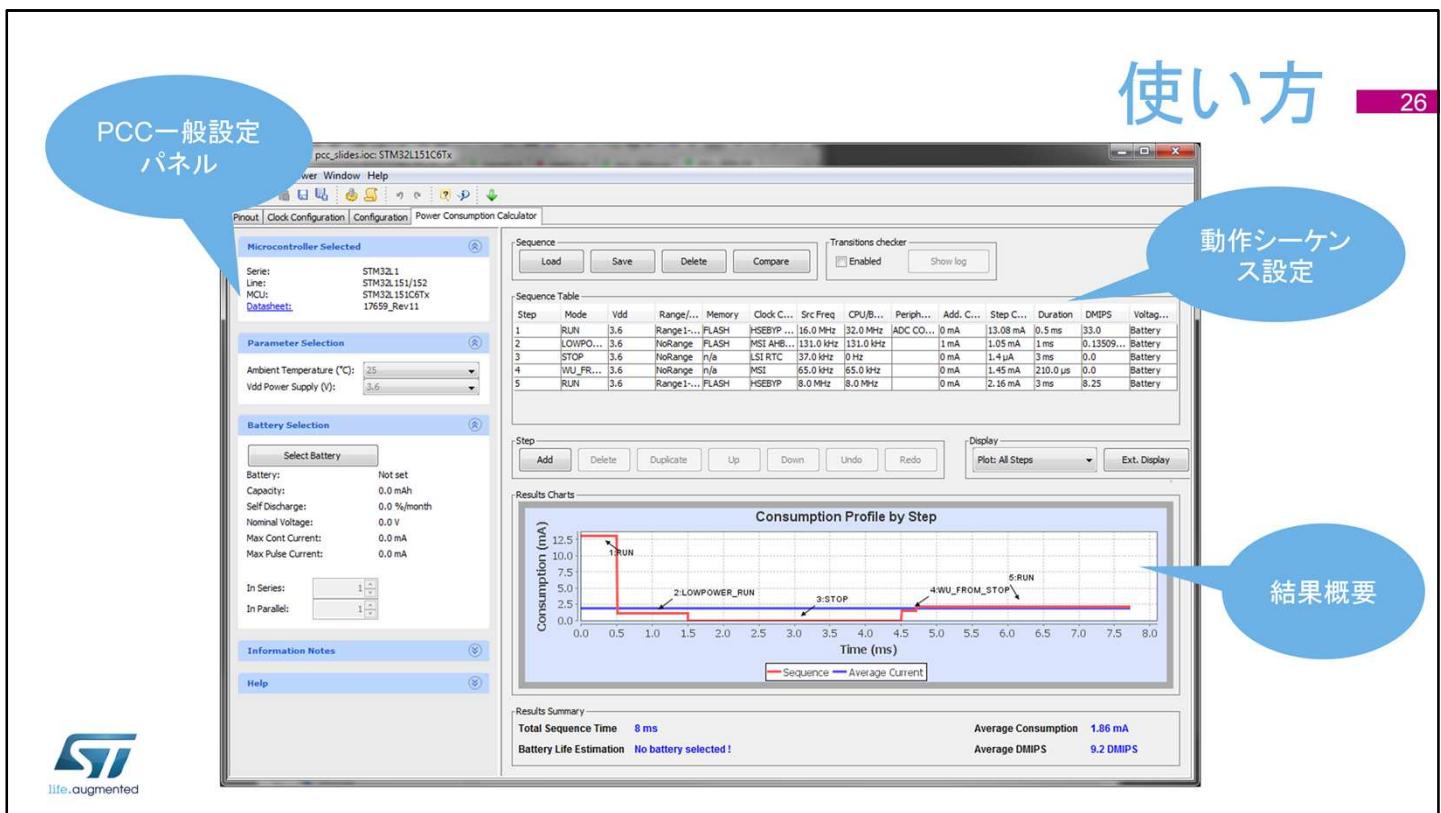
- ・電力計算機(PCC)はSTM32 マイコン群の電力消費、DMIPS、および電池寿命の見積もりに標準値のデータベースを使います。
- ・電力計算機GUIツールはSTM32CubeMXに統合されています。
- ・見積りシナリオは高度に設定可能でかつ妥当性チェック付です。
- ・電池を選択するかもしくはカスタム電池を定義します。
- ・他のマイコンや電力オプションとの比較がしやすくなっています。
- ・インポート、エクスポート、レポート生成機能があります。



電力消費計算機は主要電源もしくは補助電源として使う電池寿命の見積もりに使います。見積もりの要となるマイコンの想定動作シーケンスは簡単にインポート・エクスポートでき、不正な状態遷移も検出します。2つのマイコンについての見積もりを比較できレポートも生成できます。

# 使い方

26



電力消費計算機はSTM32CubeMXメインウィンドウの4番目のタブになります。ウィンドウは複数の区画で分割されています。

一般設定区画は現在選択中のマイコンのタイプと標準動作条件を要約します。

2番目の区画はシミュレーション動作シーケンスとその制御を表示します。

シミュレーションを実行するボタンはなく、結果がリアルタイムで表示されます。

# 一般PCCパラメータ

27

- マイコンの選択をSTM32CubeMXから継承
  - より詳細な情報を見るにはデータベースへの直リンクを使います。
- パラメータの選択
  - 選択したマイコンによっては温度・電圧の選択範囲が制限されます。
- 電池選択 – 標準のものを選ぶか新たに定義するか
  - 電池は容量、電圧、自己放電、および電流制限によって定義されます。
- 注釈情報
  - 見積制限についての注意喚起を目的とします。

**Microcontroller Selected**

Serie: STM32L1  
Line: STM32L151/152  
MCU: STM32L151C6Tx  
Datasheet: 17659\_Rev11

**Parameter Selection**

Ambient Temperature (°C): 25  
Vdd Power Supply (V): 3.6

**Battery Selection**

Select Battery

Battery: Not set  
Capacity: 0.0 mAh  
Self Discharge: 0.0 %/month  
Nominal Voltage: 0.0 V  
Max Cont Current: 0.0 mA  
Max Pulse Current: 0.0 mA

In Series: 1  
In Parallel: 1

**Information Notes**

Help



一般PCC設定区画は選択したマイコンおよびデフォルト電力源を要約しています。

温度、電圧といったパラメータは選択中のマイコンおよび利用可能な電力消費データに応じて定義されます。

電池選択区画は電池タイプの選択もしくは定義に使用します。電池源はオプションで、定義されていれば選択中のシーケンスの各ステップに使用され、デバイスをシミュレーションします。

情報およびヘルプ区画はユーザーに有用な注釈を提供します。

# シーケンスの構築

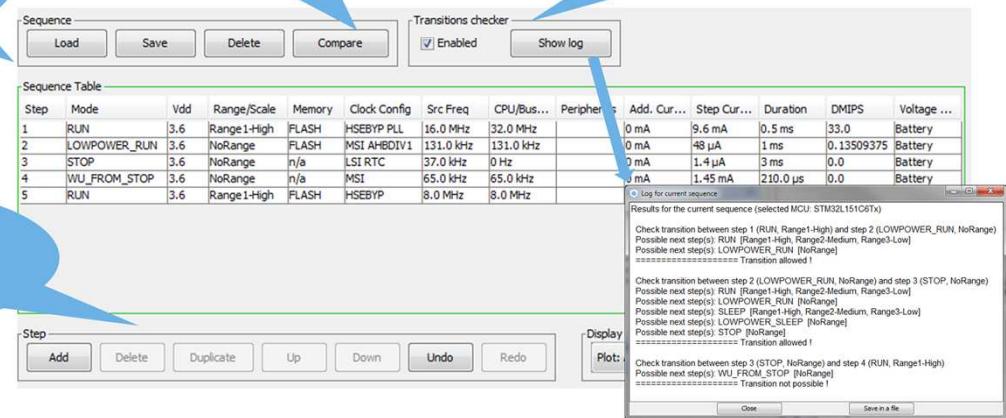
28

- シーケンスは順序付ステップの集合です。

既存のシーケンスをロードし調整します。

異なるマイコン同士であってもシーケンスを比較します。

設定された電力ステップの遷移が妥当かどうか自動でチェックします。



既存のステップの追加もしくは複製で新規ステップを生成します。

“Sequence Table”は様々な期間と設定でステップ列を定義します。その長さに実質制限はありません。

シーケンスをロード、変更、再利用できます。

ユーザーインターフェースを使うことでシーケンス内部で各ステップを複製、再配置できます。

有効していれば全ての状態遷移が自動でチェックされ、周波数もしくは電力範囲の不正な変化を防止します。問題のあるステップはシーケンステーブル上で自動でハイライトされます。

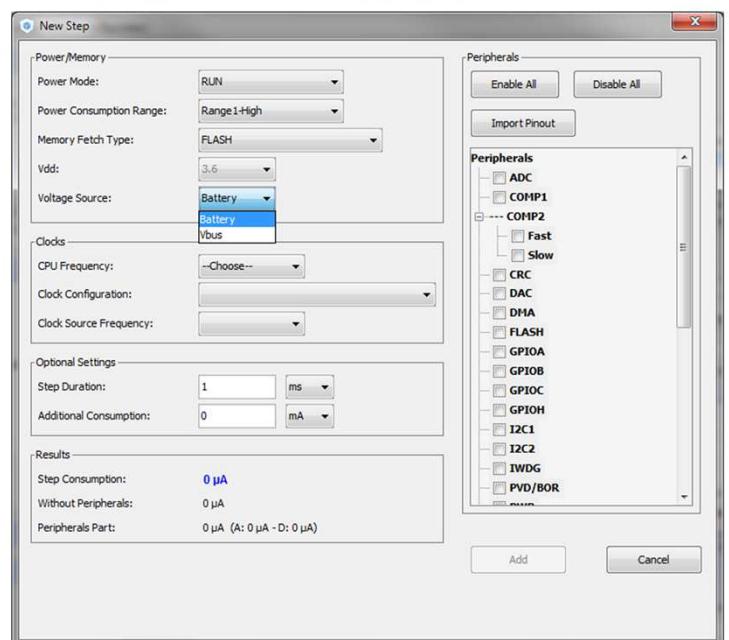
詳細な解説を表示する際は”Show log”をクリックします。

“Compare”は保存済のシーケンスが構成する現在のシナリオについて電力と性能の比較を表示します。

# 電力消費ステップの定義 (1/2)

29

1. 電力モードの選択で使用可能なペリフェラルを決定します。
2. 調整設定で性能と電力消費のバランスを調整します。
3. コードが実行されるメモリ、およびプリフェッチとバスのオプションを選択します。
4. Vdd – 実際よりもPCC上の選択肢は限られています。
5. このオプションは電池寿命の計算のために用意されています。



電力ステップをダイアログウィンドウで追加もしくは編集します。遷移チェックが有効になっていれば許容値とともに新しいステップがあります。

Power mode selection has the most significant impact on the availability of other settings, interfaces and power/performance balance.

電力ステップは複数の特性によって定義され、電力モードが最も重要な特性です。各電力モードの使用可否および特性はリファレンスマニュアルやデータシートに記載されています。

他の設定、インターフェース、および電力/性能バランスの使用可否に関して電力モードが最も影響します。

電圧レギュレータで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、電力消費は大きく低減されます。詳細についてはデータシートをご覧ください。

命令をフェッチするアドレスや関連する設定も電力消費や使用可能なクロックスピードに影響します。

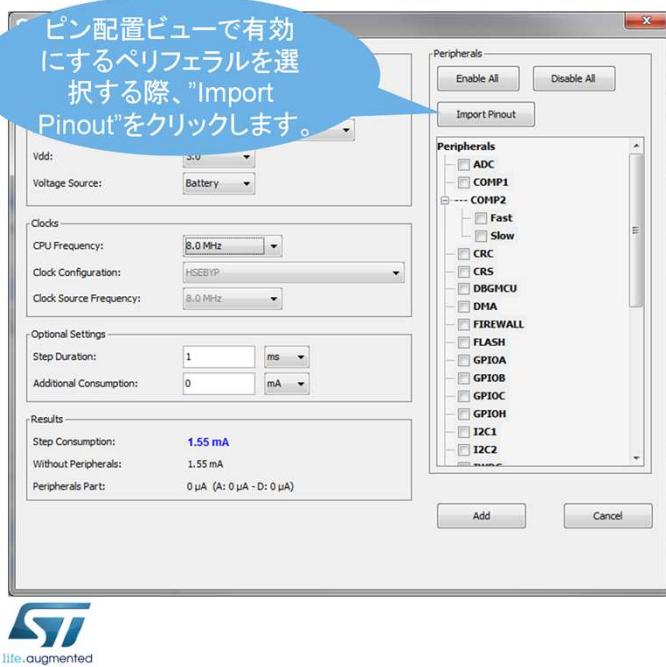
供給電圧が計算されます。実際の電圧が使えない場合は可能な限り近い電圧値を使用します。

バッテリドレンモデルでデバイスがUSBに接続されている場合などのケースでもうひとつオプションがあります。

電力モードについて更に学習するには、システム電力制御トレーニングプレゼンテーション(system power control module training presentation)をご覧ください。

## 電力消費ステップの定義 (2/2)

30



### ・クロック

- 電力消費範囲を制限するため周波数を選びます。
- 利用できるクロック設定は現在使用できるデータ、および他の設定に依存します。

### ・ペリフェラル

- ペリフェラルへのクロック入力を選びます。
- pinoutタブからクロック入力設定をインポートします。

### ・オプション設定

- 追加の電力消費はピン負荷の見積もりで表示されます。

クロック設定は電力設定、メモリ設定、さらに利用できる測定データによって制限されます。オプション一覧はドキュメントをご覧ください。

使用していないペリフェラルのクロックを無効にすると消費電力を低減できます。各ステップ期間中に使用するペリフェラルをご確認ください。"Import"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全てのペリフェラルが選択されます。

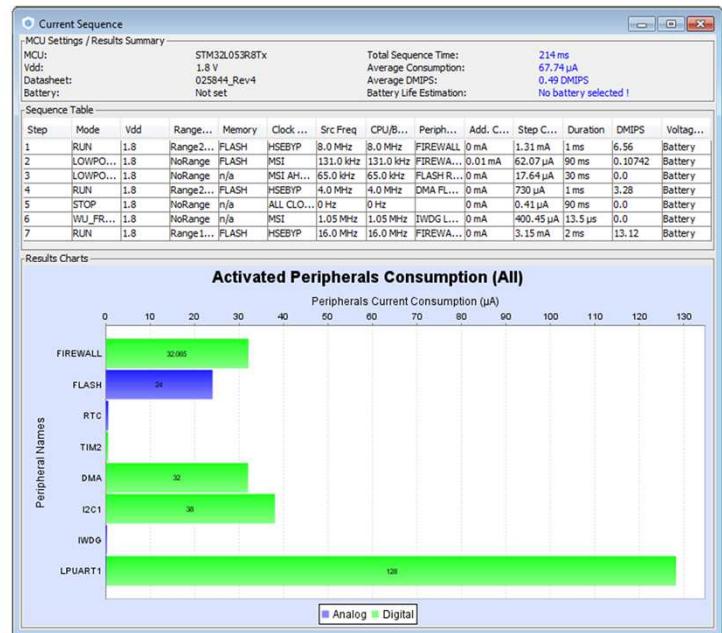
最後に各ステップ期間と追加の電力消費設定を定義します。追加の電力消費とは、ピンに接続されている様々な負荷を指します。例えばLED、ボタン、通信インターフェースなどが該当します。

# シーケンス電力消費プロファイルの表示

31

- 電力消費グラフを切り離して表示できます。

- 複数のビューを選べます。
  - 電力対時間によるプロット
  - 円グラフ
  - ペリフェラルの電力消費



電力計算機能は有効なプレゼンテーションツールです。”Ext. Display”をクリックすると電力消費プロファイルを別のウィンドウで表示できます。電流消費見積もりを様々な方式でプロットし、表示できます。デフォルトでは電力ステップシーケンスと電力消費を時間軸で表示します。

他に複数のモードで消費した電力の割合を円グラフ表示できます。その円グラフは各モードの電力消費に占める割合を示すか、もしくはRunおよび低消費電力モードの2種類に分けて表示します。

ペリフェラルの電力消費を分けてグラフ表示でき、それによって各ペリフェラルに必要な電力がわかります。表示のしかたは3種類あり、それぞれデジタルペリフェラルのみ、アナログペリフェラルのみ、もしくはその両方になります。

# レポートの生成と出力

32

2. Power Plugin report					
2.1. Microcontroller Selection					
Serie	STM32L1				
Line	STM32L151/152				
MCU	STM32L151C6Tx				
Datasheet	17659 Rev11				
2.2. Parameter Selection					
Temperature	25				
Vdd	3.6				
2.3. Sequence					
Step	STEP1	STEP2	STEP3	STEP4	STEP5
Mode	RUN	LOWPOWERSTOP	WU FROM RUN	RUN	STOP
Range	Range1-High	NoRange	NoRange	NoRange	Range1-High
Fetch type	FLASH	FLASH	n/a	n/a	FLASH
Clock Config	HSEBYP PLL	MSI	LSI RTC	MSI	HSEBYP
Clock Source Freq	16.0 MHz	131.0 kHz	37.0 kHz	65.0 kHz	8.0 MHz
CPU Freq	32.0 MHz	131.0 kHz	0 Hz	65.0 kHz	8.0 MHz
Periph.	ADC				
	COMP1				
	DAC DMA				
	GPIOA				
	GPIOH				
	IWDG RTC				

Page 2



- PDFレポートを生成するオプションがあります。
- PDFレポートは電力消費計算機なしでも生成できます。
- 完全なプロジェクトは下記のファイルで構成されます:
  - Project.ioc
  - Project.pcs
  - Project.pdf
  - Project.txt
  - Project.jpg
  - ... そしてサポートする開発環境向けのプロジェクト

拡張子.iocのファイルは静的初期化設定を記録しています。電力シーケンスは拡張子.pcsのファイルに記録されています。PDFレポートが生成されると簡略化されたテキストとピン配置のJPEGファイルも合わせて生成されます。

- さらに詳しい情報は下記の文書をご覧ください。
  - UM1718 – ユーザーマニュアル
  - DB2163 – 製品仕様書
  - TN0072 – 製品テクニカルノート
  - RN0094 – 製品リリースノート
- 本ツールはST社のウェブサイト[www.st.com](http://www.st.com)からダウンロードしてください。



STM32CubeMXコード生成ツール使用時のさらに詳しい情報については、上記文書をご覧ください。また、本ツールはst.comからダウンロードできます。

ご静聴ありがとうございました。

**IMPORTANT NOTICE – PLEASE READ CAREFULLY**

STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST's terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers' products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2015 STMicroelectronics – All rights reserved

