



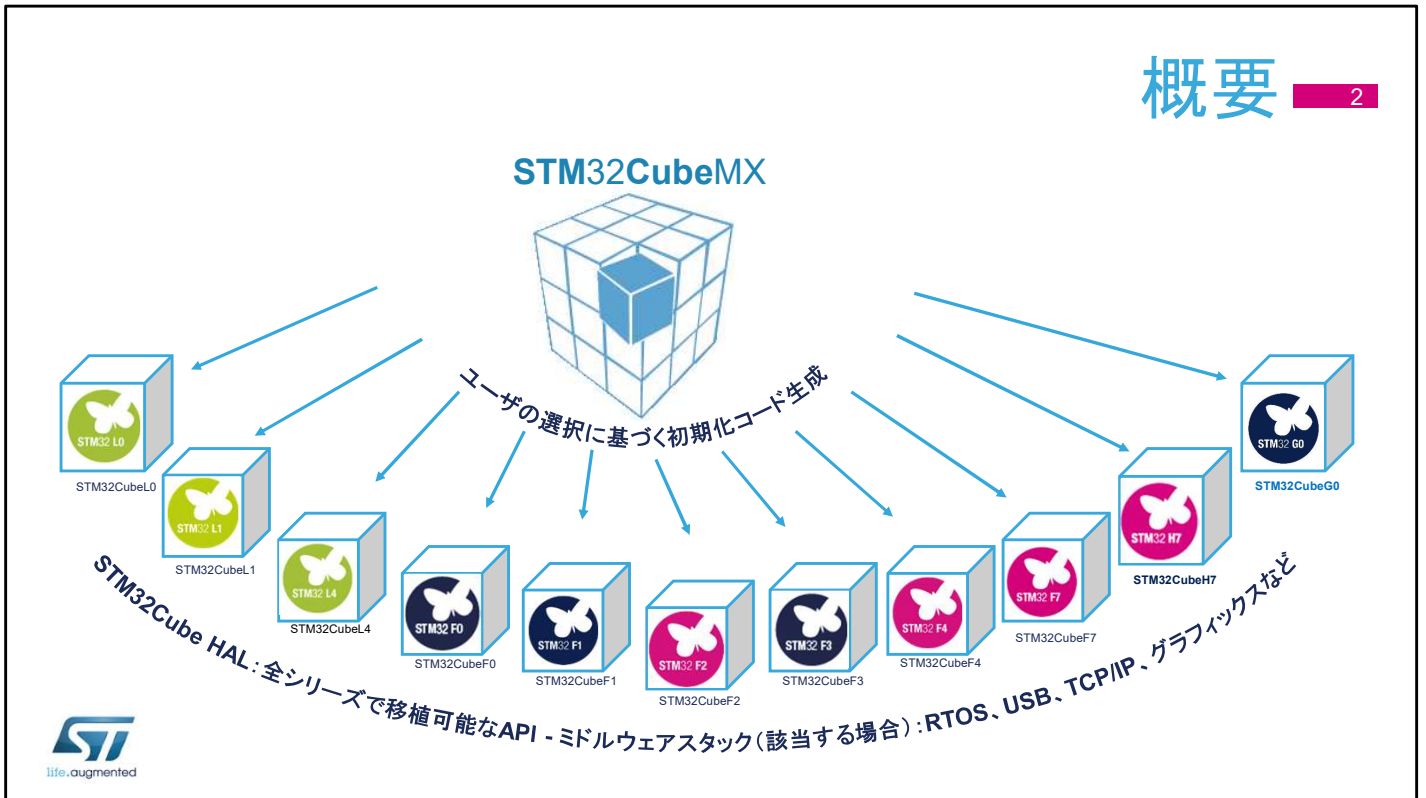
STM32G0 - STM32CubeMX

STM32CubeMXグラフィカル・ソフトウェア設定ツール

レビジョン 1.0



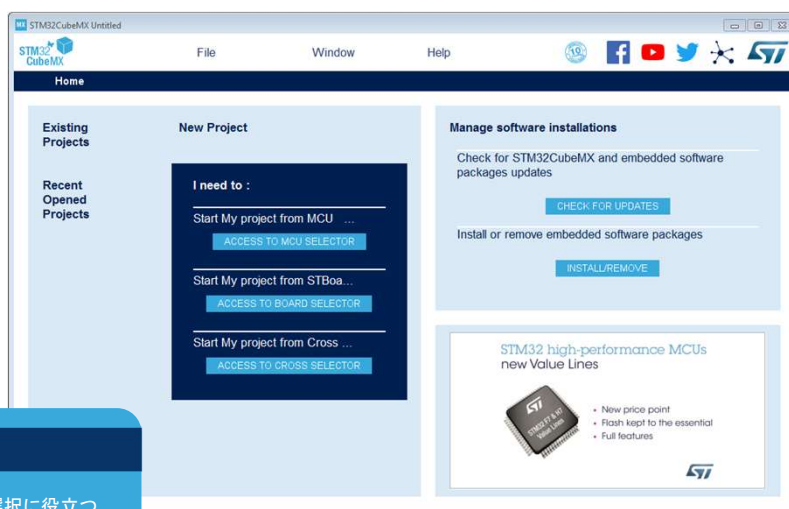
STM32CubeMX コード生成ツールのプレゼンテーションへようこそ。ここでは、このツールの主な機能、すなわちコードの設定と生成、コンパイルとデバッグ、および STM32 マイクロコントローラファミリの消費電力の推定について説明します。



これは、具体的には STM32G0 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 ファミリー全体に共通のプラットフォームです。

- 理想的なマイクロコントローラを選択してシンプルに設定

- ピン配置
- 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



アプリケーション側の利点

- 特定の目的に適したマイクロコントローラを選択に役立つ
- シミュレーションは設計フェーズに利益をもたらす
- すぐれたツールで開発速度を向上



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設定を支援するユーザインタフェースを通じて、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。ピンの割当て、クロックツリー、および内蔵ペリフェラルを設定し、生成されるプロジェクトの消費電力をシミュレーションする手段を提供します。STM32 マイクロコントローラポートフォリオの豊富なデータライブラリを使用します。開発者が機能と消費電力に関して最適な製品を選択することによって、開発の初期フェーズを容易にすることを目的としています。

- ペリフェラルと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・コードを残したままでコードを再生成可能
- コマンド・ラインとバッチ操作のオプション
- プラグインにより拡張可能
- MCUセレクト
 - ファミリ、パッケージ、ペリフェラル、またはメモリ・サイズでフィルタリング
 - 類似製品を検索
- ピン配置設定
 - 使用するペリフェラルを選択し、GPIOとオルタネート機能をピンに割当て
- NVICとDMAの設定
- クロック・ツリーの初期化
 - オシレータを選択し、PLLとクロック分周器を設定



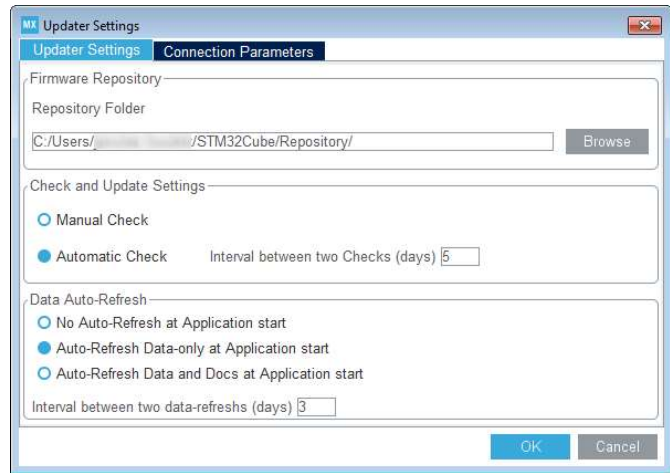
ユーザインタフェースは、適切なマイクロコントローラを選択し、必要なペリフェラルを選択し、ピン設定を割り当てるといった通常のワークフローを中心に構築されています。

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することや、場合によってはユーザコードを残したまま再生成することもできます。

前提条件と設定

5

- STM32CubeMXを使用するにはJava REが必要
 - 特定のバージョンのリリースノートで追加の要件を確認
 - Windows、Linux、およびmacOSで動作するマルチプラットフォーム・ツール
- インストール後にAlt+Sを押して、アップデートのGUIとCube FWライブラリを設定
- SWライブラリの配置場所を選択する



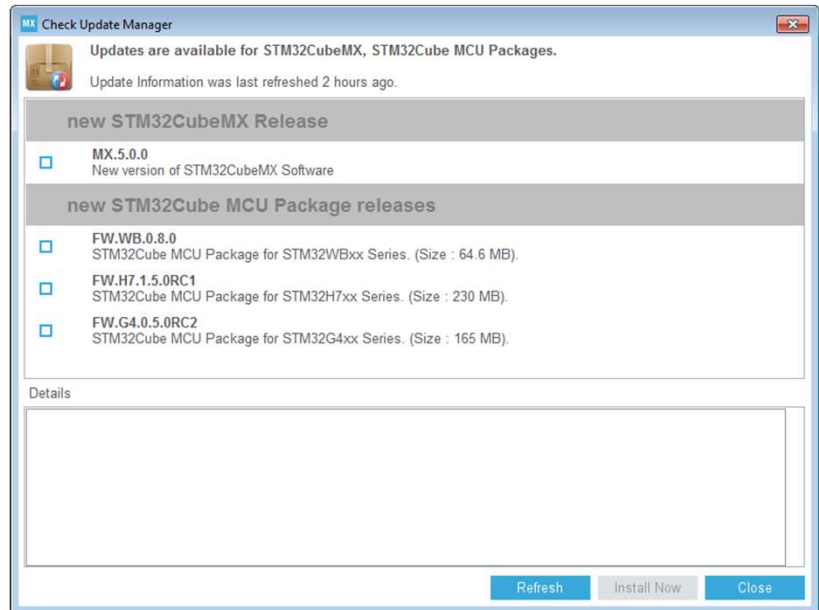
STM32CubeMX インストーラを当社の Web サイトから無償でダウンロードしてインストールします。

次に、[Settings] メニューで以下を設定します。

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

このセットアップの完了後は、新しいプロジェクトを作成できます。

- [Help] メニューから更新にアクセス可能
- ツール・アップデートで、ツールの新しいリリースおよび関連するCubeライブラリを検出可能
- ライブラリ・マネージャを使用して新しいライブラリ・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設定されている場合、ツールは、それ自体および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。

他の STM32Cube ライブラリをダウンロードする場合または相互運用性を維持するために古いバージョンを取得する場合、[Install new libraries] オプション (Alt+U) を使用します。

ただし、STM32CubeMX ツールはすべての過去のライブラリリリースでテストされているわけではなく、ライブラリの新しいリリースはツールの古いバージョンでは正しく動作しない可能性があることに注意してください。

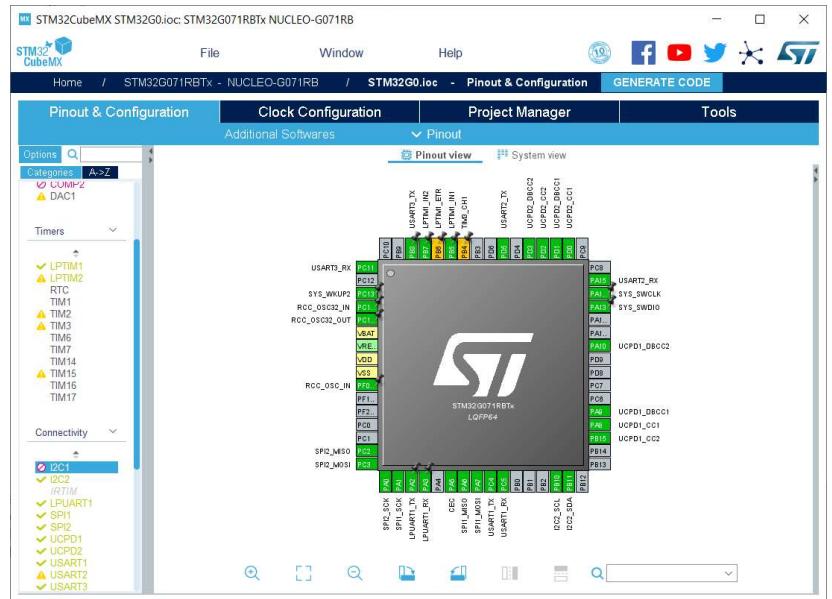
- 名前を指定してMCUを検索
 - シリーズとラインを指定する高速検索
- アプリケーション・ニーズからMCUを検索
 - パッケージ(ピン数)
 - RAMサイズ
 - NVメモリ要件
 - 組込みペリフェラル
 - インタフェースの数と種類
 - コアと周波数
 - 価格
- ドキュメントへの便利なリンク
- 表をExcelファイルにエクスポート可能

Part No.	Board	Package	Flash	RAM	ID	Freq.	GFX Score
STM32G070KB		LOFP32	128 Kbytes	36 Kbytes	30	64 MHz	0.0
STM32G070RB	NUCLEO-G070RE	LOFP64	128 Kbytes	36 Kbytes	89	64 MHz	0.0
STM32G071C8		LOFP48	32 Kbytes	36 Kbytes	44	64 MHz	0.0
STM32G071C8		UFOFPN48	32 Kbytes	36 Kbytes	44	64 MHz	0.0
STM32G071C8		LOFP48	64 Kbytes	36 Kbytes	44	64 MHz	0.0
STM32G071C8		UFOFPN48	64 Kbytes	36 Kbytes	44	64 MHz	0.0
STM32G071C8		LOFP48	128 Kbytes	36 Kbytes	44	64 MHz	0.0
STM32G071C8		UFOFPN48	128 Kbytes	36 Kbytes	44	64 MHz	0.0
STM32G071EB		WLCS25	128 Kbytes	36 Kbytes	23	64 MHz	0.0
STM32G071G5		UFOFPN28	32 Kbytes	36 Kbytes	26	64 MHz	0.0
STM32G071G8		UFOFPN28	64 Kbytes	36 Kbytes	26	64 MHz	0.0
STM32G071G8		UFOFPN28	64 Kbytes	36 Kbytes	26	64 MHz	0.0
STM32G071G8		UFOFPN28	128 Kbytes	36 Kbytes	26	64 MHz	0.0
STM32G071G8		UFOFPN28	128 Kbytes	36 Kbytes	26	64 MHz	0.0
STM32G071K8		LOFP32	32 Kbytes	36 Kbytes	30	64 MHz	0.0
STM32G071K8		UFOFPN32	32 Kbytes	36 Kbytes	30	64 MHz	0.0
STM32G071K8		LOFP32	64 Kbytes	36 Kbytes	30	64 MHz	0.0
STM32G071K8		LOFP32	64 Kbytes	36 Kbytes	30	64 MHz	0.0
STM32G071K8		UFOFPN32	64 Kbytes	36 Kbytes	30	64 MHz	0.0
STM32G071K8		UFOFPN32	64 Kbytes	36 Kbytes	30	64 MHz	0.0



[New Project] オプションを選択すると、[MCU Selector] ウィンドウが表示されます。ユーザが使用する MCU を決めている場合は、すぐに見つけることができます。決めていない場合、特定の要件を満たす製品をフィルタリングできます。

- ピン配置:
 - ペリフェラル・ツリーから作成
 - 手動で作成
- 自動で信号を再配置
- ペリフェラル／ミドルウェア (FatFS、USBなど) の間の依存関係の管理



次のステップでは、使用するペリフェラルを選択し、該当する場合はそれらの入出力にピンを割り当てます。
 独立した GPIO も設定できます。
 信号はデフォルトのピンに割り当てられていますが、別の位置 (CTRL キーを押しながらピンをクリックすると表示される) 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な場合、ツールは自動的にそれをデフォルトのピンに割り当てます。
 その際、管理しているペリフェラルとソフトウェアコンポーネントの間のほとんどの結合を自動的に考慮します。

ピン割当て(続き)

9

The screenshot shows the STM32CubeMX Pinout & Configuration window. The central area displays a pinout diagram for the STM32G071R8Tx NUCLEO-G071RB. The pins are color-coded: green for available and orange for unavailable. Callouts provide the following information:

- Callout 1 (Left):** パリフェラルは使用不可であり、すべての代替ピンが別の位置に割り当てられている (Peripheral is unusable, and all alternative pins are assigned to other positions).
- Callout 2 (Top Left):** オレンジは、パリフェラルは有効ではなく、ピン割当てのみ行われていることを意味する (Orange means the peripheral is not enabled, only pin assignment is performed).
- Callout 3 (Top Right):** ピンをクリックするとオルタネート機能が表示される (Clicking a pin displays the alternate function).
- Callout 4 (Bottom Right):** ピンアイコンを使用して信号配置を固定する (Use pin icons to fix the signal configuration).

The ST logo and 'life.augmented' are visible in the bottom left corner of the screenshot.

オルタネート機能で予約されるピンが増えると、他のパリフェラルの設定に残される選択肢が減ります。

この制限は、他のパリフェラルノードのアイコンの変化によって示されます。

ピンを左クリックすると、そのピンのオルタネート機能を表示できます。

ピンを右クリックすると、ピン割当ての名前を指定または選択できます。

特定のパリフェラルを有効にしないでピン配置を選択した場合またはピン配置に問題がある場合、ピンは緑ではなくオレンジで表示されます。

- ペリフェラルのモードが取り得るさまざまな状態
 - 淡色:別のモードが設定されている必要があるので、このモードは使用不可
 - 黄:このモードは制限付きで使用可能
 - 赤:このモードに必要な信号をピン配置にマップ不可
- 信号はピン配置ビューで直接設定／移動可能
 - ピンをクリックし、表示される使用可能な信号のリストから1つを選択
 - 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、別の位置にドラッグ
 - 未使用のピンは、コード生成機能によって省電力アナログ・モードに設定でき、無視が可能。



ペリフェラルのモードは、さまざまな状態になる可能性があります。

- 淡色:別のモードが設定されている必要があるので、このモードは使用できません。淡色モードの上にマウスポインタを移動すると、理由が表示されます。クロックソースを無効にする必要があるか、または他のペリフェラルに依存している可能性があります。
- 黄:一部のオプションが競合によりブロックされているので、このモードは制限付きで使用できます。たとえば、USARTを同期モードに設定することはできません。なぜなら、選択可能なクロックピンはすべて使用されているからです。
- 赤:このモードに必要な信号をピン配置にマップできません。たとえば、重要な信号の代替ピンがすべて他のペリフェラルによって使用されている場合に、この状況が発生する可能性があります。信号は、ピン配置ビューで直接設定／移動できます。
- ピンをクリックし、表示される使用可能な信号のリストから1つを選択します。ペリフェラルが割り当てられていないGPIOでは、この方法を使用できます。
- 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。そして(CTRLキーを押しながら)信号をドラッグして新しいピンにドロップできます。
- 手動で未使用のピンをすべてアナログに設定する必要はありません。半自動で行う手順があります。

ペリフェラルとミドルウェアの設定

11

- 使用中のペリフェラルとミドルウェアのグローバルビュー

- 設定エラーをハイライト

- + 未設定
- ✓ OK
- ⚠ ノンブロッキングの問題
- ✗ エラー

GPIO設定が正しくないと見なされているが、コードは生成されている可能性がある

クリックしてDMAを設定

[Pinout view]と[System view]のクイックな切り替え

設定にエラーがあり、コード生成機能から警告メッセージが表示

この設定は有効



メインウィンドウの [Configuration] タブでは、STM32CubeMXでセットアップできる、設定可能なすべてのハードウェア/ソフトウェアコンポーネントの概要が表示されます。

設定オプションにアクセスできる各ボタンが、設定状態を示す小さなアイコン付きで表示されます。

デフォルト状態は設定されていません。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。

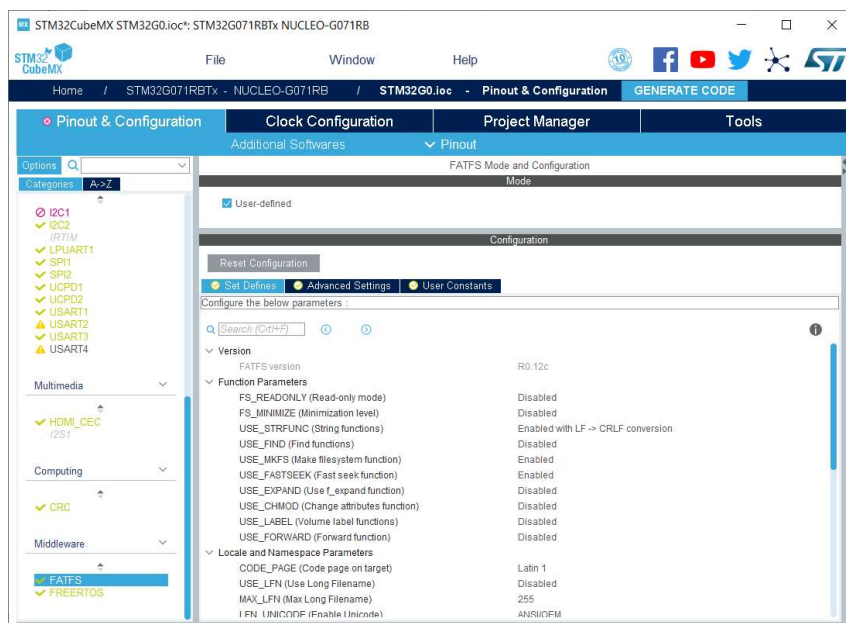
正しく設定されている場合も、さらに変更することができます。

警告記号は設定が正しくないことを通知するものであり、コードがこの状態で生成された場合、ペリフェラルは動作しません。

重要なエラーは赤の X で表され、続行するには設定を変更する必要があります。

ペリフェラルとコンポーネントをさらに追加するには、[Pinout view] タブに戻ります。

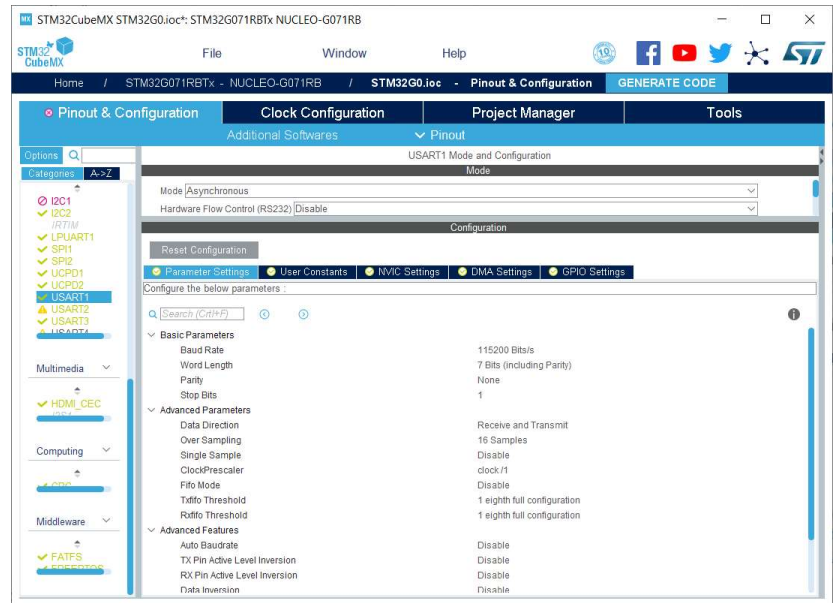
- サポートされている各ソフトウェアコンポーネントに固有のオプションを表示
- すべての設定が論理グループに編成
- クイック・リファレンスとして説明と制約が提供



life.augmented

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、すべて同じ形式で表示されるので、初期化オプションに簡単にアクセスでき、有益な情報を得られます。

- 使用可能なすべての初期化パラメータが、簡単な説明とオプションとともに表示
- 割り込みをペリフェラルに割り当てるが可能
- 該当する場合はDMAを関連付けるが可能
- ペリフェラルのGPIOの入力/出力を設定



ペリフェラルを設定する場合、ダイアログウィンドウに基本パラメータ、依存関係、および制約が表示されます。必要に応じて簡単なドロップダウンメニューを使用します。

割り込み優先順位を設定できるのは、[NVIC settings] タブのみです。ペリフェラルウィンドウで設定できるのは、各割り込みを有効または無効にすることだけです。

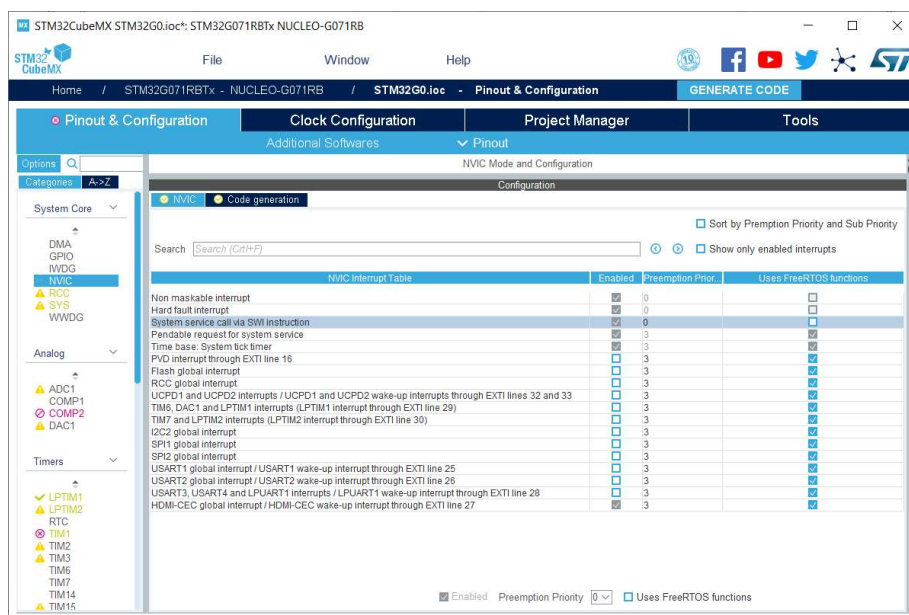
[DMA settings] タブには、DMA リクエストの初期化関連のすべてのパラメータが表示されますが、実行時関連のパラメータ（開始アドレスなど）はここでは管理しません。

[GPIO settings] タブでは、GPIO のパラメータと機能、ピンフィルタリング、および識別しやすくするために各信号にラベルを付けることができるかどうかを定義します。

NVICの設定パネル

14

- すべての割込みを一元的に制御するパネル
- 優先順位とサブ優先順位を管理
- リストの割込みを検索、フィルタリング、および並べ替え
- [Code generation] タブで割込み初期化をカスタマイズ可能

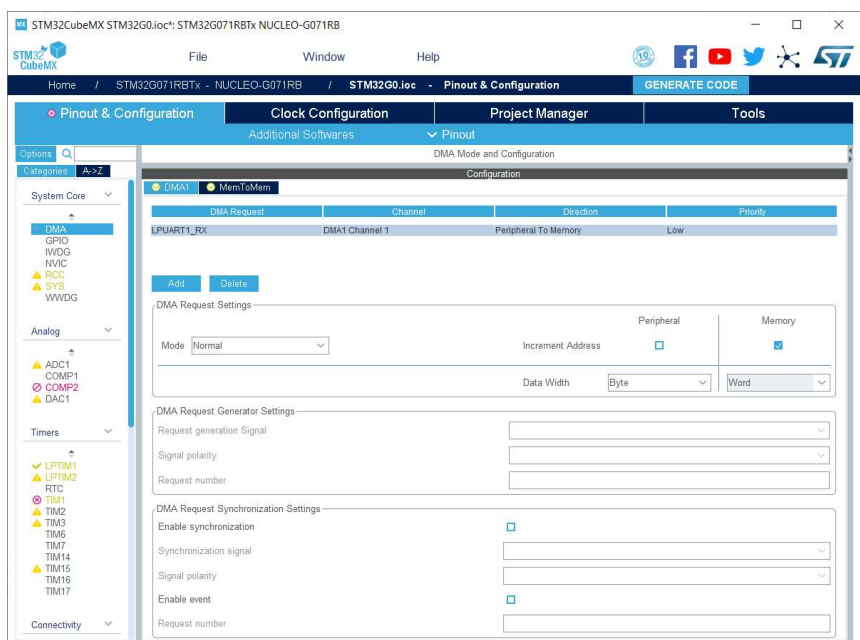


NVIC の設定パネルは、使用可能／有効な割込みの概要とそれらの優先順位を一元的にわかりやすく表示しており、STM32CubeMX の利点の 1 つです。このウィンドウでは、選択したペリフェラルの割込みを有効にしたり、それらの優先順位を設定したりします。

DMAの設定パネル

15

- メモリ間を含むすべてのDMAリクエストを管理
- 方向、優先順位などを設定する



対応する DMA チャンネルのタブを選択し、[Add] ボタンをクリックすると、指定したペリフェラルへの DMA リクエストが追加されます。すべての設定オプションを確認してください。これは DMA チャンネルを設定しますが、DMA 転送をすべて記述しているわけではないことに注意してください。すべてを記述するのは、アプリケーションコードで行う必要があります。

GPIOの設定パネル

16

- アプリケーションは、ほとんどのGPIOパラメータに実用的なデフォルト値の設定が可能
- デフォルト値は、ロースピードおよびプルアップなしとして選択
- 複数のピンを選択して同じ設定可能



[Pin configuration] ウィンドウの [GPIO] タブでは、各ピンの設定と初期化設定を容易に実行できます。

各ピンは、ピン設定の概要とそのユーザラベルとともに表形式でリストされます。

選択したピンは、ドロップダウンメニューを使用して、並べ替え、検索、および変更の適用を行います。

ツールによって割り当てられたデフォルト値は安全ですが、特定のペリフェラル設定では動作しない可能性があります。

ツールによって選択された GPIO 速度がペリフェラルの通信速度に対して十分な値であること、および必要に応じて内部プルアップが選択されていることを確認します。

設定をすばやく割り当てるには、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する方法を試してください。タブを使用すると、特定のペリフェラル専用のピングループが表示されます。

初期化中に適用される設定を実行時に変更することができますが、それは STM32CubeMX ツールの範囲外です。

- すべての初期化コードをCで生成
- サポートされている任意の開発ツールチェーン用のプロジェクト・ファイルを生成
- 専用セクション内にユーザ・コードを追加でき、このコードは再生成しても維持
- 再生成時に最新バージョンのライブラリを使用するか、同じバージョンのままにするかを選択するオプションあり

```

124  /* Private user code -----
125  /* USER CODE BEGIN 0 */
126
127  /* USER CODE END 0 */
128
129  /**
130   * @brief The application e
131   * @retval int
132   */
133  int main(void)
134  {
135      /* USER CODE BEGIN 1 */
136
137      /* USER CODE END 1 */
138
139      /* MCU Configuration-----
140
141      /* Reset of all peripherals, Initializes the Flash inte
142      HAL_Init();
143
144      /* USER CODE BEGIN Init */
145
146      /* USER CODE END Init */
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
    
```



すべての入力、出力、およびペリフェラルが設定されると、コードを生成する準備は完了です。

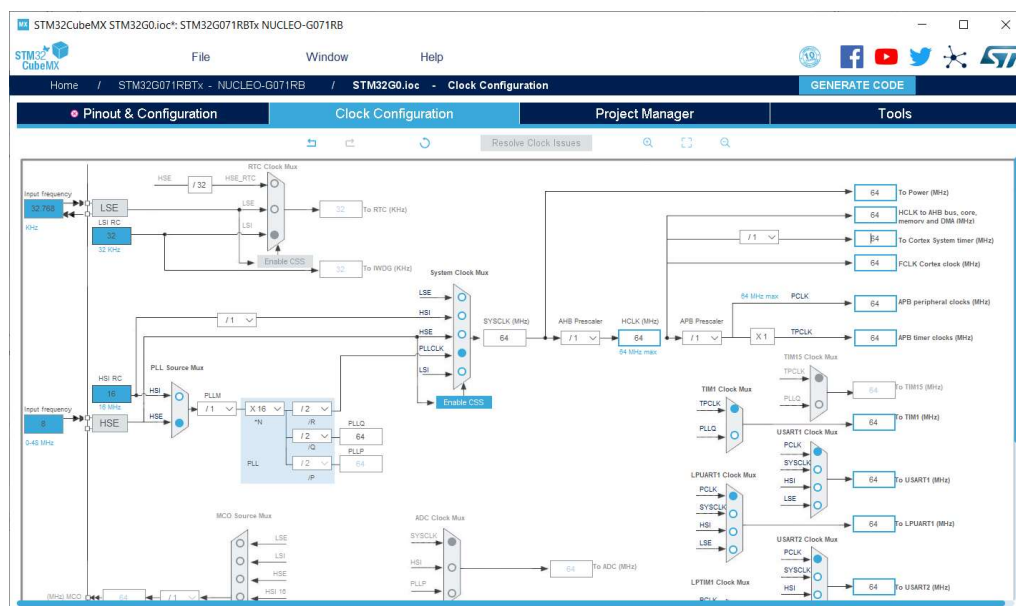
最初に、メインウィンドウの [Project] メニューで設定をチェックします。

Keil、IAR、および Atollic のツールチェーンなど、サポートされているさまざまな開発ツールの 1 つを選択して、生成されるプロジェクトを引き継ぐことができます。

カスタムコードに影響を及ぼすことなく STM32Cube MX を使用して初期化設定を変更するには、ユーザコードを "USER CODE" コメントブロックの制約範囲内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、この後のスライドを参照してください。

- すべてのクロックの値を直接表示
- 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を区別
- クロックの制約と機能を管理

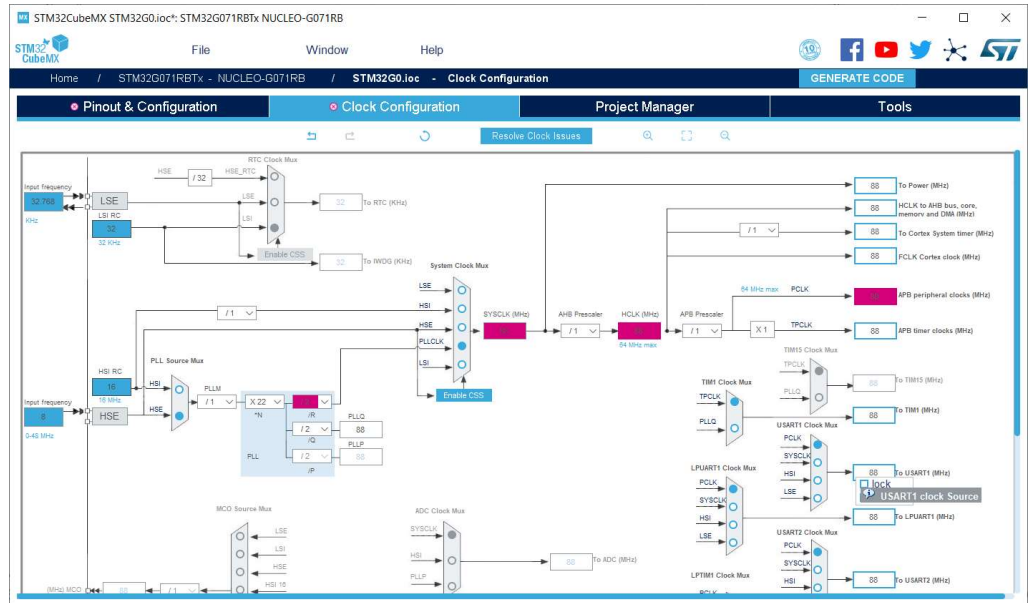


[Clock Configuration] タブには、クロックパスの概要図が表示されます。この図には、すべてのクロックソース、分周回路、および逡倍回路が含まれます。実際のクロック速度が表示されます。

アクティブで有効なクロック信号は青でハイライトされます。ドロップダウンメニューとボタンを使用して、実際のクロック設定を変更します。

クロックの設定(続き)

- エラーは発生すると同時に赤でハイライト
- 値を青枠内に入力して、ツールで分周回路と逡倍回路を調整
- ロック機能で、設定値の変更を固定



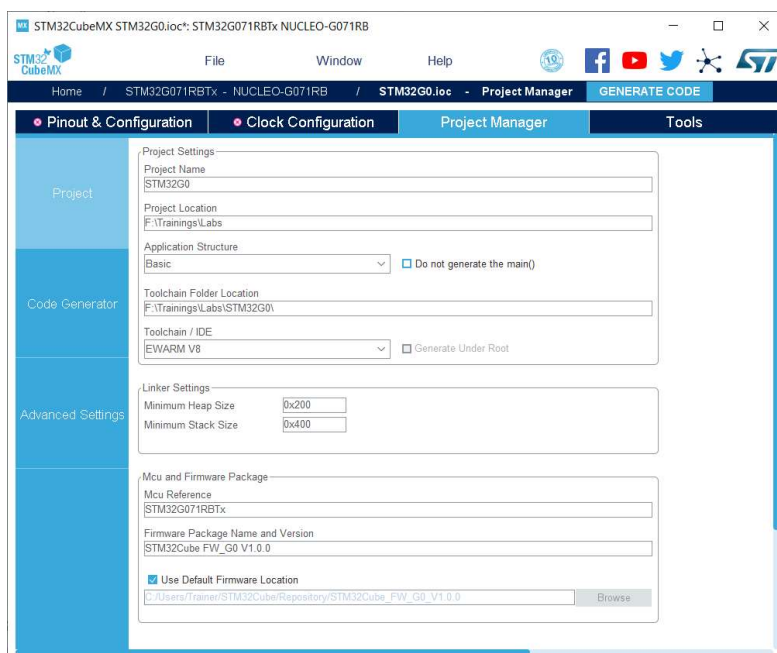
設定値が範囲外の場合、直ちに赤で表示して問題を強調します。

その逆も可能です。クロック速度の目的値を青枠内に入力すると、ソフトウェアはその速度が実現するように分周回路と逡倍回路の再設定を試みます。青のクロック値を右クリックしてロックすると、変更できなくなります。

コード生成プロジェクトの設定

20

- プロジェクトを保存する際にプロジェクト名を設定
- プロジェクトの場所を参照
- 好みのツールチェーンを選択
- 正確なマイクロコントローラの種類とライブラリ・バージョンを確認

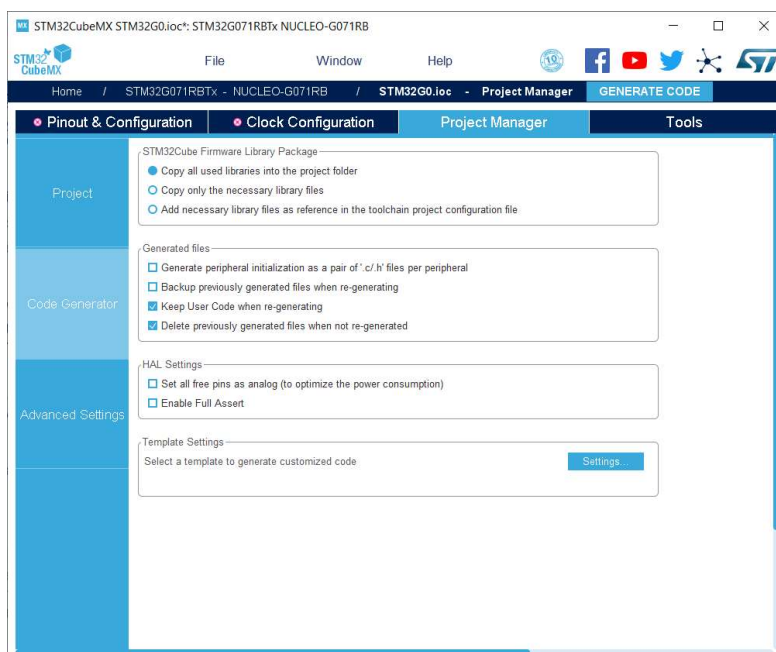


このウィンドウは、プロジェクトを保存する際([Save as...])に表示されます。ツールチェーンフォルダとは、実際のツールチェーンアプリケーションの場所ではなく、ツールチェーンのワークスペースが配置される場所を指します。プロジェクト設定を表示するために Alt+P ショートカットを使用して、このダイアログウィンドウを制限付きバージョンで表示することも可能です。

コード生成のオプション

21

- ライブラリ・パッケージ
 - ライブラリの全体または必要な部分だけを生成されるプロジェクト・フォルダにコピー可能
 - またはライブラリを当初の場所に置いたまま、すべてのプロジェクトから参照
- [Generated files]
 - 各ペリフェラルを個別のファイルまたは共通ソースファイルで初期化
 - 古いファイルを使用するためのオプション
 - ユーザ・コードをそのまま残すオプションはここで設定
- [HAL Settings]
 - 使用可能なピンをアナログに設定すると消費電力が減少するが、**ピン配置でSWD/JTAGを明示的な選択は慎重に**
 - フルアサート機能はデバッグに有効



STM32Cube HAL 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ることができます。プロジェクトを小型パッケージとして移行する必要があ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、[Copy ~] オプションを選択します。ライブラリを当初の場所に置いたままにすると、複数のプロジェクトでライブラリの最新バージョンを共有しやすくなります。

また、すべてのペリフェラルの初期化コードをまとめて

stm32fxxx_hal_msp.c ファイル内に生成すること、またはペリフェラルごとに1つずつファイルを生成することができます。

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、推奨するワークフローの問題です。これらのオプションは再生成機能に関連することに注意してください。またここでは [Keep User Code when re-generating] オプションを有効にします。

[Set all free pins as analog] 設定は消費電力の削減に役立ちますが、[Pinout] タブで SWD/JTAG インタフェースが明確に選択されていない場合、このオプションによりデバッグインタフェースが無効になります。

[Enable Full assert] により、HAL 関数に渡されるパラメータのチェックが有効になり、デバッグに過大な労力を注がなくてもユーザコードのいくつかのバグを明らかにするのに役立つ可能性があります。

警告および免責事項

22

- このツールは、STM32ファミリ全体を全般的に支援するものなので、特定の製品の個々の機能にまで対応が行き届かないことがある
- **STM32CubeMX GUIツールは、リファレンス・マニュアルまたはデータシートに代わるものではない**
 - 詳細については、常にドキュメントを参照すること
 - 重要な機能は、製品または HAL にはあるが、GUI にはないことが多い
- GUIは、プロジェクトを開始し、動作する開始設定を初期化するのに役立つ。ただし、この設定は実行時に動的に変更することができる(GPIO、NVIC の優先順位、またはクロック設定)。



ユーザインタフェー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を全般的に支援します。ただし、多様化する STM32 のポートフォリオの概要を使いやすい形で表示すると同時に、各製品の詳細にすべて対応することはできません。

不明な点がある場合は、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、もっと詳細で正確な情報を入手してください。また、知識を深めるために、ぜひアプリケーションノートやサンプルをお読みください。プロトタイプを速やかに動作させるために STM32CubeMX を使用してアプリケーションを開始し、動的な変更が必要になったとき(通常は、同じアプリケーションで別のクロックまたは GPIO の設定をサポートするため)にコードを変更するのが、一般的な方法です。

ユーザが STM32CubeMX ジェネレータを使用して定義されたユーザ領域内にコードを記述していれば、ユーザインタフェースのトップレベルでいくつかの変更を適用する必要がある場合に STM32CubeMX の初期セットアップに戻すことが可能です。この変更では、通常は GPIO ピン設定の追加、クロックの変更、または NVIC の優先順位の変更などを行います。



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、通常は低消費電力を主な設計目標にします。データシートから消費電力レベルを抽出するのは、時間がかかる面倒な作業です。消費電力計算機能は、この作業を簡素化するために、データシート値をスマートユーザインタフェースツールに抽出し、設定可能なシナリオから有益な推定値を生成します。

- 消費電力計算機能(PCC)は、標準値のデータベースを使用して、STM32マイクロコントローラの消費電力、DMIPS、およびバッテリー寿命を推定
- GUI ツールはSTM32CubeMXに統合
- シナリオは詳細に設定可能で有効性チェック機能付き
- バッテリー選択機能を使用するか、またはカスタム・バッテリーを定義
- 他のマイクロコントローラまたは他の電力オプションとの比較が安易
- レポートのインポート、エクスポート、および生成



消費電力計算機能を使用して、主電源または補助電源として使用するバッテリーの寿命を推定できます。シーケンスは簡単にインポート／エクスポートできます。不正な状態遷移も検出されます。異なる2つのマイクロコントローラ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。

Step	Mode	V _{DD}	Range/Scale	Memory	CPU/Bus Freq	Caps Config	Peripherals	Step Current	Duration
1	RUN	3.0	Range+High	FLASH	64 MHz	HSE BYP PLL		6.25 mA	1 ms
2	SLEEP	3.0	Range+High	FLASH	64 MHz	HSI PLL		1.85 mA	1 ms
3	RUN	3.0	Range+High	SRAM1 Flash PowerDo	48 MHz	HSI BYP PLL		4.9 mA	1 ms
4	LOWPOWER_RUN	3.0	NoRange	SRAM1 Flash PowerDo	1 MHz	HSI Regulator_LP		225 µA	1 ms
5	STOP0	3.0	NoRange	Flash PowerDown	16 MHz	HSI		100 µA	1 ms

PCC一般設定
パネル

シーケンス設定

結果の概要

消費電力計算機能は、STM32CubeMX メインウィンドウの 4 番目のタブです。このウィンドウは複数のペインに分割されています。

一般設定ペインには、標準的な動作条件および現在選択されているマイクロコントローラのタイプの概要が表示されます。

2 番目のペインには、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なく、結果がすぐに表示されます。

PCCの一般的なパラメータ

26

- マイクロコントローラの選択 (STM32CubeMXから取得)
 - データシートへの直接リンクを使用して詳細情報を取得
- パラメータの選択
 - 選択するマイクロコントローラによっては、温度と電圧の選択肢が制限される可能性有り
- バッテリーの選択 – 既存のものを選択するか、独自に定義
 - バッテリーは、容量、電圧、自己放電、および電流制限によって定義
- 情報ノート
 - 目的は推定値の制限についての警告

STM32G071RBTx

Series	STM32G0
Line	STM32G0x1
Datasheet	DS12232_Rev0

T_A 25°C / V_{DD} 3.0V

Li-MnO2(CR1225) (1x1)

In Series	1	In Parallel	1
Capacity	48.0 mAh		
Self Discharge	0.12 %/month		
Nominal Voltage	3.0 V		
Max Cont Current	1.0 mA		

Information Notes

Help



life.augmented

PCC 一般設定ペインは、多くの情報を提供し、選択したマイクロコントローラとデフォルトの電源ソースの概要を表示します。選択したマイクロコントローラおよび入手できる消費電力データによっては、温度、電圧などのパラメータも定義できます。バッテリー選択ペインは、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。オプションのバッテリーソースは、定義されている場合はシーケンスの選択されているステップでのみ使用でき、単独および外部電源ソースに接続された状態の両方で動作するデバイスをシミュレーションします。情報セクションとヘルプセクションには、ユーザにとって役に立つ注意事項が表示されます。

シーケンスの構築

27

- シーケンスは一連の順序付きステップ

ステップの追加または既存ステップの複製により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応

シーケンスを比較 (異なるマイクロコントローラでも可能)

提案された電力ステップ遷移が有効かどうかを自動的にチェック

The screenshot shows the ST Studio IDE interface. At the top, there are three buttons: 'Step', 'Sequence', and 'Transitions Checker'. Below them is a 'Sequence Table' with the following data:

Step	Mode	V _{DD}	Range/Scale	Memory	CPU/Bus Freq	Clock Config	Peripherals	Step Current	Duration
1	RUN	3.0	Range 1-High	FLASH	64 MHz	HSE BYP PLL	ADC11fs_10_Msps AHB_APB...	7.19 mA	1 ms
2	SLEEP	3.0	Range 1-High	FLASH	64 MHz	HSI PLL	BusMatrix FLASH	2.11 mA	1 ms
3	RUN	3.0	Range 1-High	SRAM1 Flash-PowerDown	48 MHz	HSI EYP PLL	ADC11fs_2.5_Msps AHB_APB...	5.98 mA	1 ms
4	LOWPOWER_RUN	3.0	NoRange	SRAM1 Flash-PowerDown	1 MHz	HSI Regulator_LP	ADC11fs_2.5_Msps AHB_APB...	885.52 μA	1 ms
5	STOP0	3.0	NoRange	Flash-PowerDown	16 MHz	HSI	IWDG*	100 μA	1 ms
6	WU_FROM_STOP0	3.0	NoRange	Flash-PowerDown	16 MHz	HSI		1.3 mA	5.5 μs

At the bottom left, there is the ST logo and the text 'life.augmented'.

[Sequence Table] では、一連のステップをさまざまな所要時間や設定とともに定義します。その長さには事実上制限はありません。

シーケンスは、ロード、変更、および再利用することができます。ユーザインタフェースでは、個々のステップを複製したり、シーケンス内で位置を変更したりできます。

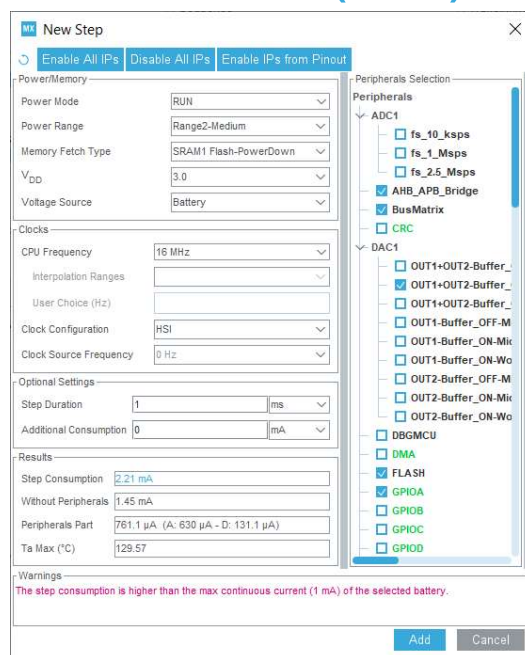
有効な場合、基本的な妥当性ルールに照らしてすべての状態遷移をチェックして、周波数／電力レンジの不正なジャンプを防ぎます。問題のあるステップは、[Sequence Table] ですぐにハイライトされます。

[Show log] 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、現在表示されているシナリオの電力と性能を、保存されているシーケンスと比較した結果を表示します。異なるマイクロコントローラなど、異なる設定間で比較して評価することもできます。

消費電力ステップの定義 (1/2)

1. 電力モードの選択によって、ペリフェラルの使用を決定
2. レギュレータの設定により性能と消費電力のバランスを取る
3. コードを実行す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
4. Vdd - 代表的な設定を複数用意
5. このオプションはバッテリー寿命を計算するために存在



このダイアログウィンドウでは、電力ステップを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な場合、許容される値が新しいステップにプリセットされます。

電力ステップは、最も重要な電力モードを筆頭に、さまざまな特性によって決まります。各電力モードの可用性と特性については、個々の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。電力モードの選択は、他の設定の可用性、インタフェース、および電力／性能のバランスに最も重大な影響を及ぼします。

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は制限されますが、多くの場合、消費電力が大幅に削減されます。詳細については、データシートを参照してください。

命令をフェッチするアドレスと関連する設定も、消費電力と使用可能なクロック速度に影響を及ぼします。

消費電力の計算に使用する供給電圧は、実際の電圧がわからない場合は、可能な限り近い値が使用されます。

最後のオプションは、たとえばバッテリー消費モデルでデバイスが USB に接続されているケースを除外するために存在します。

電力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。

消費電力ステップの定義 (2/2)

29

• クロック

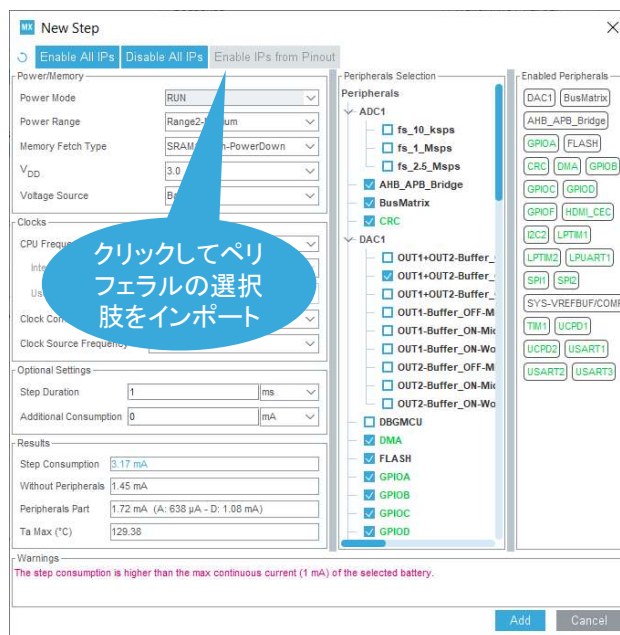
- 周波数の選択肢は消費電力レンジによって制限
- 使用可能なクロック設定は使用可能なデータと他の設定に依存

• 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
- ピン配置タブから選択肢をインポート

• オプション設定

- 他の消費電力は推定されるピンの負荷によって表示



クロック設定は、電力とメモリの両方の設定および使用可能な測定データによって制限されます。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。

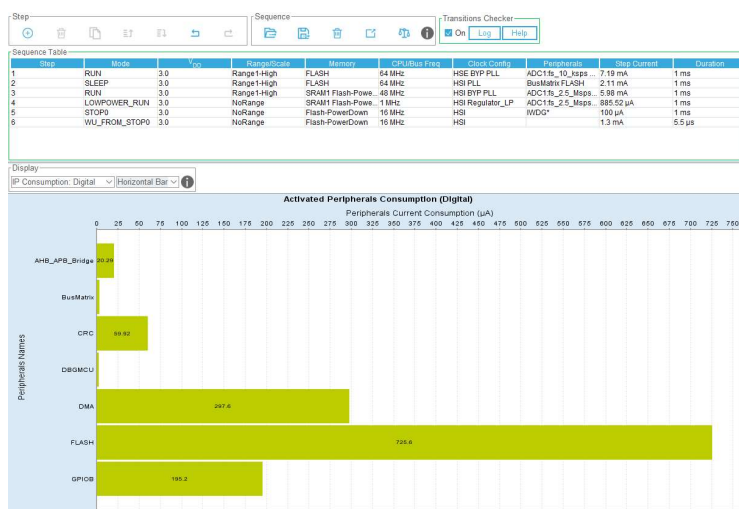
未使用のペリフェラルのクロックを無効にすることは、エネルギーを節約する確実な方法です。ステップ所要時間中に電源が供給されるペリフェラルをチェックします。[Import] ボタンをクリックすると、生成されるコードで初期化されるすべてのペリフェラルが選択されます。

最後に、ステップ所要時間と他の消費電力設定を定義します。他の消費電力は、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。

シーケンスの消費電力プロファイルの表示

30

- グラフを分離してプレゼンテーション目的で外部ディスプレイに出力可能
- 複数の異なるビューを選択可能
 - 電流と時間の関係をプロット
 - 円グラフ
 - ペリフェラルの消費電力



life.augmented

消費電力計算機能は、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、別ウィンドウでレポートが表示されます。消費電流推定値をグラフィック形式でプロットする多数の異なる方法が用意されています。デフォルトは、電力ステップシーケンスと経時的な消費電力に基づく方法です。

別の方法として、さまざまなモードの消費エネルギーの割合のグラフを作成できます。円グラフで各モードの占有率を表示する方法や、分割して RUN モードと低消費電力モードのみ表示する方法があります。

ペリフェラルの消費電力を分割して、グラフに電力要件をプロットすることもできます。デジタルペリフェラルのみ、アナログペリフェラルのみ、または両方が混在するビューをプロットできます。

出力およびレポートの生成

31

- オプションのステップとしてPDFレポートを生成
- PDFレポートは、PCCを使用しなくても生成可能
- 保存されるプロジェクト・ワークの内容：
 - 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

STM32G0 Project
Configuration Report

6. Power Consumption Calculator report

6.1. Microcontroller Selection

Series	STM32G0
Line	STM32G0x1
MCU	STM32G071RBTx
Datasheet	DS12232 Rev0

6.2. Parameter Selection

Temperature	25
Vdd	3.0

6.3. Battery Selection

Battery	Li-MnO2(CR1225)
Capacity	48.0 mAh
Self Discharge	0.12 %/month
Nominal Voltage	3.0 V
Max Cont Current	1.0 mA
Max Pulse Current	5.0 mA
Cells in series	1
Cells in parallel	1



拡張子 .ioc のファイルには、静的な初期化設定が含まれます。電力シーケンスは、拡張子 .pcs で保存されます。PDF レポートが生成され、簡略化されたテキストおよびピン配置図を含む別個の JPG 画像も一緒に生成されます。

- 詳細については、次のリソースを参照：
 - UM1718: ユーザマニュアル
 - DB2163: 製品仕様
 - TN0072: STM32 ST-LINKユーティリティの製品テクニカルノート
 - RN0094: 製品リリースノート
- ツールはSTのWebサイト www.st.com からダウンロードする



STM32CubeMX コード生成ツールの使用方法の詳細については、このスライドに示されているドキュメントを参照してください。これらのドキュメントは、www.st.com からダウンロードできます。