

Embedded Code Generation *DEMO MODEL*

Simple PIL Model

シンプルなPILモデル

- TI C2000 TSPのReadおよびOverride Probeブロックを調査してPILシミュレーションを実行 -

Last updated in C2000 TSP 1.6.1

1 はじめに

このデモでは、誘導負荷に電力を供給する電流制御Hブリッジ回路を紹介します。TI C2000 Target Support Package(TSP)とTexas Instruments (TI) C2000 Microcontrollers (MCU)を使用した、PLECSのPIL(Processor-in-the-loop)ワークフローについて説明します。

このモデルを実行するには、次の製品が必要です:

- PLECS BlocksetまたはStandalone
- PLECS Coder
- TI C2000 TSP
- PLECS Processor-in-the-loop (PIL)
- サポートされているTI C2000 MCU

注意 このモデルには、以下の方法でアクセスできるモデル初期化コマンドが含まれています:

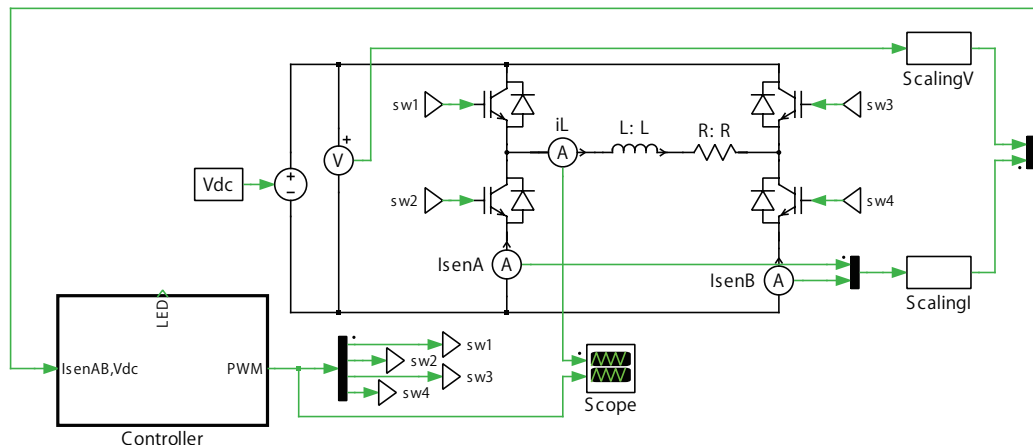
PLECS Standalone: シミュレーションメニュー -> シミュレーション・パラメータ... -> 初期化

PLECS Blockset: Simulinkモデルウィンドウで右クリック -> モデル プロパティ -> コールバック -> InitFcn*

2 モデル

最上位レベルの回路図には、[図1](#)に示すように、Controllerサブシステムとプラント回路が含まれています。Controllerサブシステムは、選択して編集メニュー -> サブシステム -> 実行の設定...からコード生成機能の有効化にチェックして有効にします。この手順は、PLECS Coderを介してサブシステムのモデル コードを生成するために必要です。

図1: モデルの最上位レベルの回路図



Hブリッジ回路とコントローラについては、TI C2000デモモデルライブラリの"Hブリッジコンバータ"で詳しく説明しています。本書は、TI C2000 TSPを使用したPILワークフローにのみ焦点を当てています。

2.1 電源回路

[図1](#)に示す電源回路には、 $V_{dc} = 24V$ のDC電圧源が供給されます。Hブリッジは、誘導負荷に電力を供給する2つのIGBTハーフブリッジパワー素子モジュールコンポーネントで構成しています。パルス幅変調(PWM)スイッチング信号は、Controllerサブシステムから取得します。スケールされたDC入力電圧と出力インダクタ電流の測定値は、Controllerサブシステムに送信されます。

2.2 Controller

Controllerサブシステムを図2に、制御ロジックを図3に示します。

"Right Leg Duty-Cycle"サブシステムは、感知した入力電圧Vdcの変動を考慮しながら、右レッグHブリッジ出力で平均12Vを維持するために必要なデューティー比を決定します。"Left Leg Duty-Cycle"サブシステムは、比例積分(Proportional-Integral: PI)コントローラに基づいて左レッグHブリッジの変調指数を決定します。検出したインダクタ電流は、-3Aと3A間で切り替わる設定点と比較されます。この誤差は、アンチwindアップロジックを備えたデジタルPIコントローラによる電流補償に使用します。スイッチング周波数は10kHzに設定されています。コントローラの離散化ステップサイズは、スイッチング周期と同じです。

図2: Controllerサブシステム

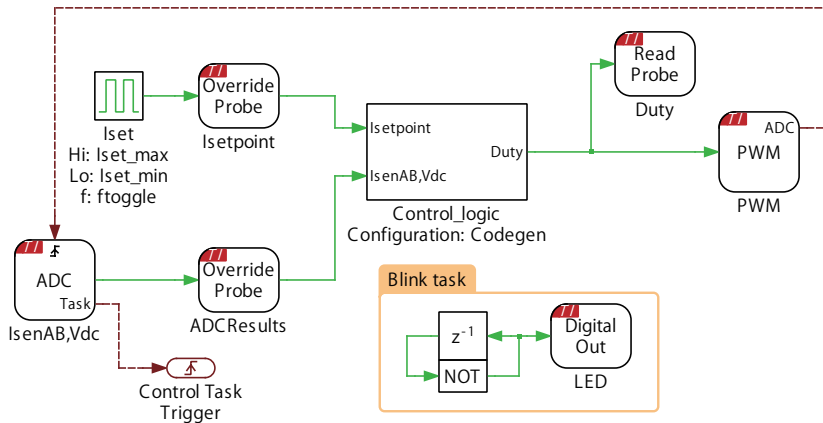
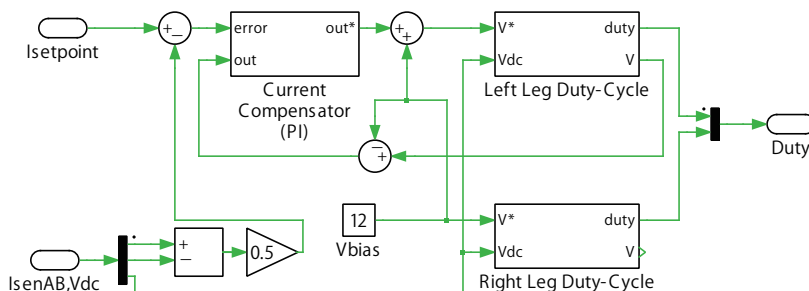


図3: Hブリッジ回路の制御ロジック



Controllerサブシステム内には、"Control_logic"という可変サブシステムがあります。2つの構成が可能です。最初の構成は"Codegen"です。この構成では、制御ロジックはPLECS Controlライブラリのコンポーネントを使用して実装します。この構成を使用して、PLECSでオフラインシミュレーションを実行することができます。2番目の構成は"PIL"です。この構成では、制御ロジックはTI C2000 MCU上で実行します。

ReadおよびOverride Probe

PILシミュレーション中に、Override Probeを使用すると、PLECSは組み込みコード内の変数を上書きできます。このモデルでは、図2に示すように、Controllerサブシステムには2つのOverride Probeが含まれています。"setpoint"というラベルのOverride Probeは、電流リファレンス設定値を変更できます。また、"ADCResults"というラベルのOverride Probeは、感知したアナログ測定値を変更できます。Read Probeを使用すると、PLECSは組み込みコード内の変数を読み取ることができます。このモデルでは、"Duty"というラベルのRead Probeが、MCUによって計算されたデューティー比を読み取り、それをPWMジェネレータに送ります。

Override ProbeとRead Probeプローブを使用するコンセプトにより、実際のMCUで実行する制御コードを、PIL用に特別に再コンパイルすることなく、PLECSシミュレーションに組み込むことができます。Override ProbeとRead Probeは、組み込みソフトウェア内に必要なだけ残しておけるテストポイントと同等のものと考えられます。必要に応じて、このようなテストポイントを持つソフトウェアモジュールを、いつでもPILシミュレーションに組み込むことができます。

PILシミュレーションの詳細についてはPIL User Manual[1]を参照してください。

3 PILワークフローとシミュレーション

このデモ モデルは、通常のPLECSシミュレーションの実行に加えて、生成された組み込みコードとPILシミュレーションを実行するように設定されています。すべてのペリフェラルブロック(ADC、PWMなど)のデフォルトのI/O構成は、TI 280039C[3]、TI280049C[4]、TI28069[5]、TI28379D[7] LaunchPad、およびTI 28388D[10] controlCARDをサポートしています。

さらに、デモモデルでは、TI 28377S [6] LaunchPad、TI 280039C [8]、TI 28379D [9] controlCARDのコード生成が可能です。これを設定するには、シミュレーションメニュー -> シミュレーション・パラメータ... -> 初期化タブからモデル初期化コマンド ウィンドウに移動し、board_typeの値を変更して、目的のボードを選択します。また、Coderオプションウィンドウで対応するターゲットとBoardタイプを適宜構成する必要があります。

PILシミュレーションを実行するには、以下の手順に従います。

1 Control_logicをCodegenに設定: 図2の"Control_logic"というラベルの可変サブシステムをCodegenに設定します。

2 MCUに書き込み: ControllerサブシステムをTI MCUにアップロードするには、以下の手順に従います。

- USBケーブルで目的のMCUをホストコンピュータに接続します。
- **Coder** -> **Coderオプション...** ウィンドウの**システム**リストから、"Controller"を選択します。
- 次に、**ターゲット**タブで、ドロップダウンメニューから適切なターゲットを選択します。次に、**General**サブタブで、目的の**Build type**を選択します。
- 次に、PLECSから直接MCUをビルドしてプログラムするには、**Build configuration**でRun from FlashまたはRun from RAMのいずれかを選択し、**Board**タイプでLaunchPadを選択して、**ビルド**をクリックします。

注意 正しくプログラムされていれば、LaunchPadボード上のLEDが点滅します。

Code Composer Studioに精通している上級ユーザ向けには、Generate code into CCS projectオプションがあります。TI C2000 Target Supportパッケージには、projectsというタイトルのフォルダが含まれています。フォルダ内には、各MCU用に事前に構築されたCCSプロジェクトを含むZIPアーカイブがあります。目的のターゲットに対応するzipアーカイブ フォルダをCCSにインポートします。CCSワークスペースに新しいプロジェクトが作成されているはずですが、CCSプロジェクトの\${workspace_loc}/dev_28xx/cg/フォルダの場所を**CCS project directory**フィールドに入力し、**Build**をクリックします。次に、通常のCCSプロジェクトとしてプロジェクトのビルドとデバッグに進みます。詳細な手順については、TI C2000 Target Support User Manual[2]の"クイックスタート"を参照してください。

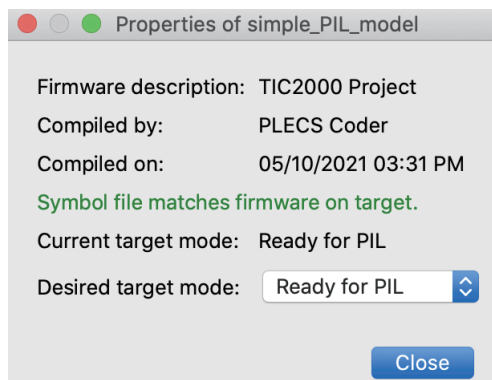
3 Symbol file: 生成された Controller.outまたはController.elfファイルを見つけます。生成された.elfファイルは、デフォルトではPLECSモデルファイルと同じディレクトリ内の_codegenというサフィックスを持つフォルダに配置されます。これは次のステップで使用する**Symbol file**です。Symbol fileは、ターゲット上で実行されるコードに対応するバイナリファイル("オブジェクトファイル"とも呼ばれます)です。

4 Control_logicをPILに設定: 図2の"Control_logic"というラベルの可変サブシステムをPILに設定します。

5 PILターゲットを構成: PILターゲットを構成するには、以下の手順に従います。

- "Control_logic"サブシステムを右クリックし、**サブシステム -> サブシステムのモデル表示**を選択します。
- PILブロックをダブルクリックしてPILパラメータウィンドウを開き、**設定**をクリックします。
- 新しいターゲットを追加するには、**+**ボタンをクリックします。必要に応じてターゲットの名前を変更してください。
- ...ボタンをクリックして、上記の**シンボルファイル**を追加します。PLECSは、PILシミュレーションのほとんどの設定、およびOverride ProbeとRead Probeのリストとその属性をシンボルファイルから取得します。
- 適切な**デバイス種類**と**デバイス名**を選択し、**確認**をクリックします。
- **プロパティ**ボタンをクリックして、[図4](#)のように、シンボルファイルがターゲットのファームウェアと一致していることを確認します。

図4: PILターゲットのプロパティ

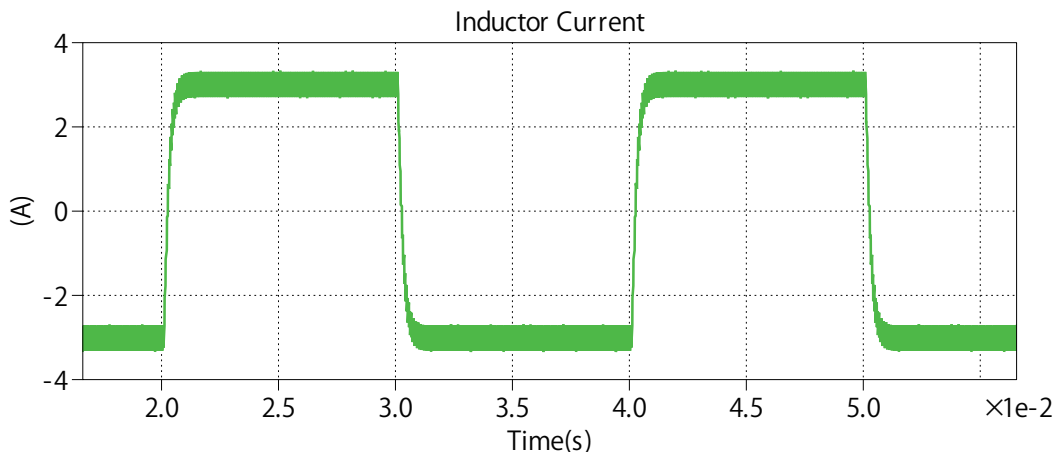


- 目的のターゲット モードをReady for PILに変更します。これは、ターゲットが PILシミュレーションの準備ができていることを意味し、これはパワーステージが無効になっている安全な状態に対応します。
- PILパラメータ ウィンドウに戻り、構成されたこの**ターゲット**を選択します。

6 PILシミュレーションを開始: 最後に、シミュレーション -> 開始からPILシミュレーションを開始します。

インダクタ電流とPWM波形はPLECSスコープで表示できます。インダクタ電流リファレンスは、Controllerサブシステムの "Iset" (パルス発生器)コンポーネントを使用して、-3Aと3Aの間で切り替えられます。インダクタ電流のステップ応答を[図5](#)に示します。

図5: TI 28069 LaunchPadを使用したPILシミュレーションによるインダクタ電流測定



4 まとめ

このモデルは、TI C2000 TSPのRead ProbeおよびOverride Probeブロックを調査して、PILシミュレーションを実行しました。また、TI C2000 TSPとTI C2000 MCUを使用したステップバイステップの手順を含むPILシミュレーションワークフローも提供しました。

5 参考文献

- [1] PIL User Manual,
URL: <https://plexim.com/sites/default/files/pilmanual.pdf>.
日本語版: <https://adv-auto.co.jp/products/plexim/manual.html>
- [2] PLECS TI C2000 Target Support User Manual,
URL: <https://www.plexim.com/sites/default/files/c2000manual.pdf>.
日本語版: <https://adv-auto.co.jp/products/plexim/manual.html>
- [3] TI C2000 F280039C LaunchPad development kit,
URL: <http://www.ti.com/tool/LAUNCHXL-F280039C>.
- [4] TI C2000 Piccolo MCU F280049C LaunchPad development kit,
URL: <http://www.ti.com/tool/LAUNCHXL-F280049C>.
- [5] TI C2000 Piccolo MCU F28069M LaunchPad development kit,
URL: <http://www.ti.com/tool/LAUNCHXL-F28069M>.
- [6] TI C2000 Delfino MCU F28377S LaunchPad development kit,
URL: <https://www.ti.com/lit/pdf/sprui25>.
- [7] TI C2000 Delfino MCU F28379D LaunchPad development kit,
URL: <http://www.ti.com/tool/LAUNCHXL-F28379D>.
- [8] TI C2000 F280039C controlCARD evaluation module,
URL: <https://www.ti.com/tool/TMDSCNCD280039C>.
- [9] TI C2000 F28379D controlCARD development kit,
URL: <https://www.ti.com/tool/TMDSCNCD28379D>.
- [10] TI C2000 F28388D controlCARD evaluation module,
URL: <https://www.ti.com/tool/TMDSCNCD28388D>.

改訂履歴:

- C2000 TSP 1.3.1 初版
- C2000 TSP 1.4.5 Webリンクを更新
- C2000 TSP 1.5.1 28388Dおよび28379D controlCARDのサポートを追加と、コントローラでの倍精度演算の使用を最小限に抑制
- C2000 TSP 1.6.1 280039C LaunchPadおよびcontrolCARDターゲットのサポートと自動ピン選択を追加

 Pleximへの連絡方法:

☎ +41 44 533 51 00 Phone

+41 44 533 51 01 Fax

✉ Plexim GmbH Mail

Technoparkstrasse 1
8005 Zurich
Switzerland

@ info@plexim.com Email

<http://www.plexim.com> Web

Advancing Automation
 アドバンオートメーションへの連絡方法:

☎ +81 3 5282 7047 Phone

+81 3 6285 0250 Fax

✉ ADVAN AUTOMATION CO.,LTD Mail

1-9-5 Uchikanda, Chiyoda-ku
Tokyo, 101-0047
Japan

@ info-advan@adv-auto.co.jp Email

<https://adv-auto.co.jp/> Web

Embedded Code Generation Demo Model

© 2002–2023 by Plexim GmbH

このマニュアルに記載されているソフトウェアPLECSは、ライセンス契約に基づいて提供されています。ソフトウェアは、ライセンス契約の条件の下でのみ使用またはコピーできます。Plexim GmbHの事前の書面による同意なしに、このマニュアルのいかなる部分も、いかなる形式でもコピーまたは複製することはできません。

PLECSはPlexim GmbHの登録商標です。MATLAB、Simulink、およびSimulink Coderは、The MathWorks、Inc.の登録商標です。その他の製品名またはブランド名は、それぞれの所有者の商標または登録商標です。