

NI VeriStand Target *DEMO MODEL*

RL Load Control

RL負荷制御

- シンプルな一次システムのループバック制御 -

Last updated in NI VeriStand TSP 1.0.1

1 はじめに

このデモモデルは、RL回路(resistive-inductive circuit)の応答をシミュレートするために使用するNI PXIeシステムを示しています。このモデルには、外乱後のRL回路における電流の流れを調整するPIコントローラも含まれています。

このモデルは"仮想プロトタイピング"プラットフォームとして構成されており、プラントモデルとコントローラの両方が同じターゲットハードウェア上でリアルタイムに実行されるため、HIL(hardware-in-the-loop)プラットフォームおよびリアルタイムコントローラとしてのハードウェアの能力を強調しています。仮想プロトタイピングを用いることで、リアルタイムのターゲットに追加のハードウェアを接続することなく、プラントモデルとコントローラモデルをテストすることが可能になります。

このモデルを実行するには、以下の製品が必要です:

- バージョン4.5.4以降の**PLECS Standalone**または**Blockset**のライセンス、および**PLECS Coder**のライセンスが必要です。
- NI VeriStand Target Support LibraryをユーザのPCにダウンロードしてインストールする必要があります。NI VeriStand Target Support User Manual[2]のクイックスタートに記載されている手順に従って、PLECSとNI VeriStandターゲットを構成してください。
- NI Linux RTおよび互換性のあるNI-DAQモジュールを備えたNI PXIまたはPXIeシステム。サポートされているモジュールの完全なリストは、VeriStand Manual[1]に記載されています。
- NI-DAQのアナログ入出力信号を接続するためのIOケーブル、ブレークアウトボード、およびアクセサリ。

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

モデルの概要を図1に示します。このモデルは、プラントとコントローラモデルの両方を含むサブシステムで構成されています。"NI PXIe"サブシステムを選択した状態で、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ます。この手順は、NI VeriStandターゲット用のモデルコードを生成するために必要です。

プラントからの入出力信号はループバック方式を用いてコントローラに接続され、またその逆も同様です。回路電流測定値*iL_plant*は、コントローラ入力*iL_control*に接続されたアナログ出力信号を表します。同様に、コントローラからの出力電圧は、プラント内の制御電圧源に接続しています。

アナログ出力信号が変化した場合、次のモデルコード実行時までには値の変化を検出しません。この遅延は、図1に示す遅れ要素コンポーネントによって明示的にモデル化されます。

図1: 信号ループバックと遅れ要素を示すトップレベルの回路図

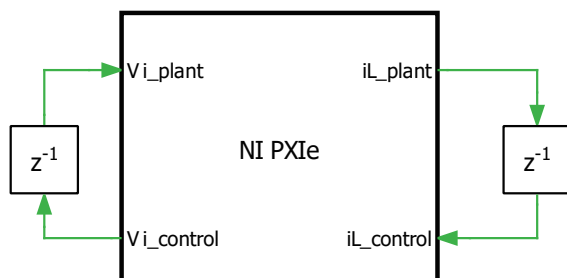
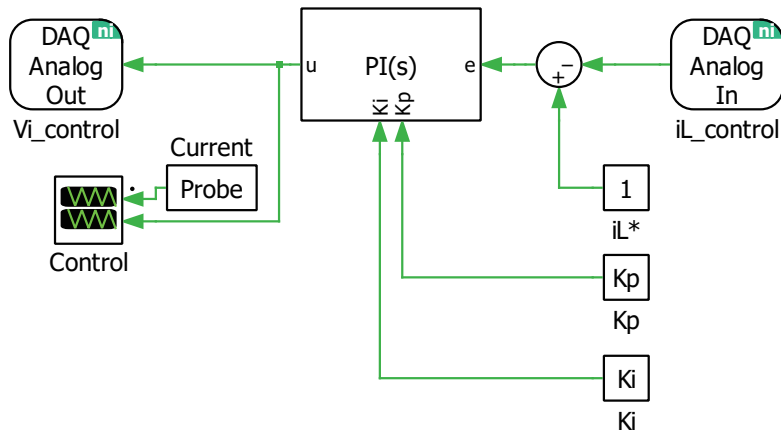
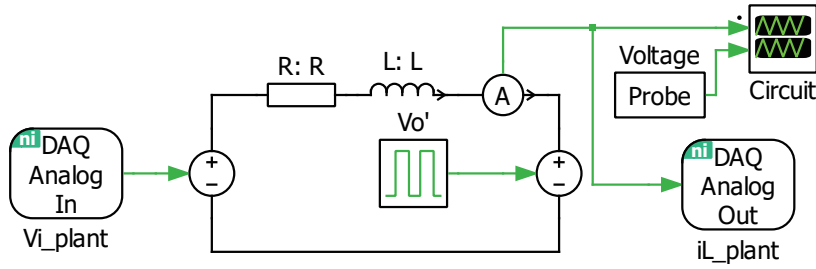


図2に示すように、電氣的プラントと回路コントロールの両方が"NI PXIe"サブシステムに含まれています。NI VeriStand Target Support Libraryのコンポーネントを使用して、NI-DAQハードウェアのアナログ出力と入力を自動的に構成します。

図2: プラントモデル(上)とコントローラ(下)の回路図



2.1 プラント

プラントは、入力電圧源と出力電圧を接続するRL回路で構成されています。両方の電圧源は可変です。パルス発生器は、プラント内の外乱をシミュレートするために、200ミリ秒ごとに出力電圧を周期的に切り替えます。回路への入力電圧を上昇させて、RL回路内の電流の流れを調整します。

プラントの伝達関数は次のように定義できます。ここで、 V_{RL} はRL回路の両端の電圧です(つまり、 $V_{RL} = V_i - V_o$)。したがって、このプラントは時定数 $\tau = \frac{L}{R}$ の実極を1つ持ちます。

$$P(s) = \frac{I_L}{V_{RL}} = \frac{1}{R + sL} = \frac{1/L}{R/L + s}$$

2.2 コントローラ

コントローラは、インダクタ電流を"iL"設定値である1.0Aに調整します。PIコントローラのゲインは、メイン回路図上の定数"Kp"と"Ki"を変更することで調整可能です。

PIコントローラの伝達関数は以下のとおりです:

$$C(s) = \frac{V_o}{I_L} = K_p + \frac{K_i}{s} = \frac{sK_p + K_i}{s} = \frac{a(s + \omega_c)}{s}$$

ループバック遅延に伴うわずかな遅延を無視した場合、結果として得られる開ループ伝達関数は次のようになります:

$$OL(s) = P(s)C(s) = \frac{1/L}{R/L + s} \frac{\alpha(s + \omega_c)}{s}$$

$\omega_c = R/L$ を選択すると極が相殺され、開ループ伝達関数は $OL(s) = \alpha/(Ls)$ となります。したがって、 α を調整することで、クロスオーバー周波数を直接選択することができます。このモデルでは、クロスオーバー周波数は初期値として $\omega_c = 2\pi 50$ rad/sに選択します。

2.3 モデルの離散化

コード生成プロセスの一環として、PLECSモデルをシステムの離散化表現に変換する必要があります。離散化ステップサイズ T_d を選択します。離散化モデルが連続システムに近似するためには、選択した T_d はプラントに関連する時定数よりも十分に小さくなければなりません。このモデルでは、離散化ステップサイズは $T_d = \tau/20$ 、つまりRL回路の時定数の $1/20^{\text{th}}$ として選択します。 T_d が増加すると、ループバック遅延に伴う遅延の影響がシステムの閉ループ応答に及ぼします。選択した T_d が大きすぎると、閉ループシステムが不安定になります。しかし、選択した T_d が小さすぎると、リアルタイムプロセッサには、次のモデルステップの前にモデルを実行するための十分な時間がなくなり、モデルステップがスキップされたり、モデルのパフォーマンスが低下する可能性があります。

3 シミュレーション

モデルをオフラインで実行するだけでなく、NI VeriStandソフトウェアを介して、またはPLECSを介して直接、NIターゲットハードウェアにモデルをデプロイすることができます。

NI VeriStand Target Supportパッケージを使用すると、PLECSはNI VeriStandにインポートできるコンパイル済みモデルを生成します。生成されるモデルには、[図2](#)に示すブロックに対応する入力信号と出力信号が含まれます。**Coder -> Coder オプション...** -> **ターゲットタブのBuild type**パラメータがVeriStand engineまたはCustom engineに設定されている場合、入力ポートと出力ポートは適切なNI-DAQハードウェアポートに自動的にマッピングされます。**Build type**がModel onlyに設定されている場合、入力ポートと出力ポートはVeriStandソフトウェア内で手動で割り当てます。

"IL"電流基準値とPIコントローラゲインは、ターゲットに接続した状態でリアルタイムに調整できるように設定されています。パラメータを調整可能にするには、**Coderメニュー -> Coderオプション...** を開き、**パラメータのインライン化**タブに移動します。回路図からコンポーネントを**例外**リストにドラッグアンドドロップすると、そのコンポーネントに関連付けられたパラメータが実行時に調整可能になります。この動作は**デフォルト設定**に依存することに注意してください。**例外**リストには、デフォルトの設定とは逆の動作をするコンポーネントが指定されているためです。

このモデルは**Build type**がModel only用に構成されていますが、すべてのビルドタイプ用にモデルを構成する方法については、以下に記載しています。ハードウェアターゲットの設定やその他のビルドオプションに関する詳細な手順は、NI VeriStand Target Support User Manual[\[2\]](#)に記載されています。

注意 対象デバイスで使用可能なハードウェアは、デモモデルで指定しているデフォルトのハードウェアI/Oと一致しない場合があります。続行する前に、NI Maxでターゲットマシン用の新しいハードウェア構成ファイル(*.nce)を生成し、モデルで使用するスロット番号とIOチャンネルを更新してください。新しいハードウェア構成ファイルの生成方法については、NI VeriStand Target Support User Manual[\[2\]](#)の**クイックスタート**を参照してください。

Model only

以下に、Model onlyのBuild typeを使用して、NI VeriStandにモデルを手動でインポートする手順を示します:

- **Coder**メニューから**Coderオプション...**を開き、ウィンドウの左側に表示されるシステムリストから"NI PXIe"を選択します。
- **ターゲット**タブに移動し、**Build type**をModel onlyに設定し、**ビルド**をクリックします。
- 生成されるモデルのデフォルトパスは、モデルファイルと同じディレクトリにあるRL_load_controlという名前です。生成されるモデルの名前は"NI PXIe"サブシステムの名前と一致し、拡張子は*.soになります。
- NI VeriStandを開き、モデルをインポートします。PLECS から生成されたモデルをインポートする手順については、NI VeriStand Manual [1] のAdding and Configuring a Modelを参照してください。
- NI VeriStand内で、使用可能なアナログ入力と出力をモデルの入力ポートと出力ポートに手動でマッピングします。

VeriStand engine

以下に、Veristand engineのBuild typeを使用して完全なVeriStandプロジェクトを自動的に作成する手順を示します:

- **Coder**メニューから**Coderオプション...**を開き、ウィンドウの左側に表示されるシステムリストから"NI PXIe"を選択します。
- **ターゲット**タブに移動し、**Build type**をVeristand engineに設定します。NIリアルタイムターゲットの設定を入力します。その際、適切な**Hardware configuration file from NI Max**を含めてください。構成ファイルの生成方法については、NI VeriStand Target Support User Manual [2]のクイックスタートセクションを参照してください。
- モデルをターゲットに自動的にデプロイするかどうかを選択(**Deploy to the target after build**)します。その場合は、NI VeriStandアプリケーションを開いてください。アプリケーションが開いたら、**ビルド**をクリックします。
- 生成されたVeriStandプロジェクトを開きます。デフォルトのディレクトリは、モデルファイルと同じ場所にあり、RL_load_control_codegenという名前です。生成されるプロジェクト名は"NI PXIe"となり、拡張子は*.nivsprjとなります。
- NIリアルタイムターゲット上でモデルを実行中に、VeriStandアプリケーションまたはPLECSを使用してモデルを操作することができます。PLECSの外部モード経由で接続するには、**Coder -> Coderオプション...**ウィンドウを開き、**外部モード**タブを選択します。**ターゲットデバイス**にlocalhostと入力し、**接続**をクリックします。

Custom engine

以下に、Custom engineのBuild typeを使用してNIリアルタイムハードウェアにモデルをデプロイする手順を説明します:

- **Coder**メニューから**Coderオプション...**を開き、ウィンドウの左側に表示されるシステムリストから"NI PXIe"を選択します。
- **ターゲット**タブに移動し、**Build type**をCustom engineに設定します。NIリアルタイムターゲットの設定を入力します。その際、適切な**Hardware configuration file from NI Max**を含めてください。構成ファイルの生成方法については、NI VeriStand Target Support User Manual [2]のクイックスタートセクションを参照してください。
- モデルをターゲットに自動的にデプロイするかどうかを選択(**Deploy to the target after build**)し、**ビルド**をクリックします。
- PLECSの外部モード経由で接続するには、**Coder -> Coderオプション...**ウィンドウを開き、**外部モード**タブを選択します。**ターゲットデバイス**のリモートターゲットのIPアドレス(例: 192.168.0.105)を入力し、**接続**をクリックします。

図3と図4は、Custom engineを使用してモデルをターゲットにビルドした場合のオフライン結果とリアルタイム結果の比較を示しています。プロットを揃えるには、**ターゲットチャンネルオプション**を5: [Vo:Source voltage]に、ターゲットレベルを2.0に設定して、**自動トリガを有効化**ボタンをクリックします。オフライン結果とリアルタイム結果は、ほぼ一致しています。選択した離散化時間 T_d を長くすると、オフラインシミュレーションとリアルタイムシミュレーションの整合性に悪影響をおよぼします。

図3: オフラインおよびリアルタイムの結果のベンチマーク

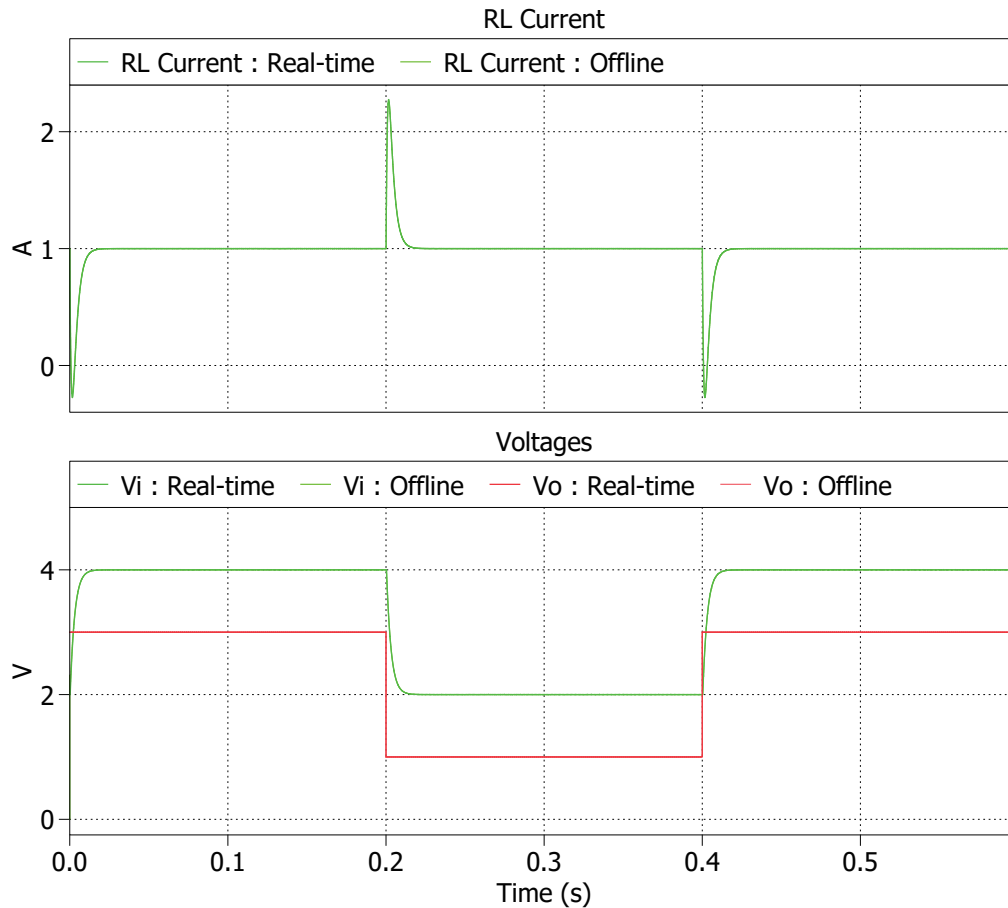
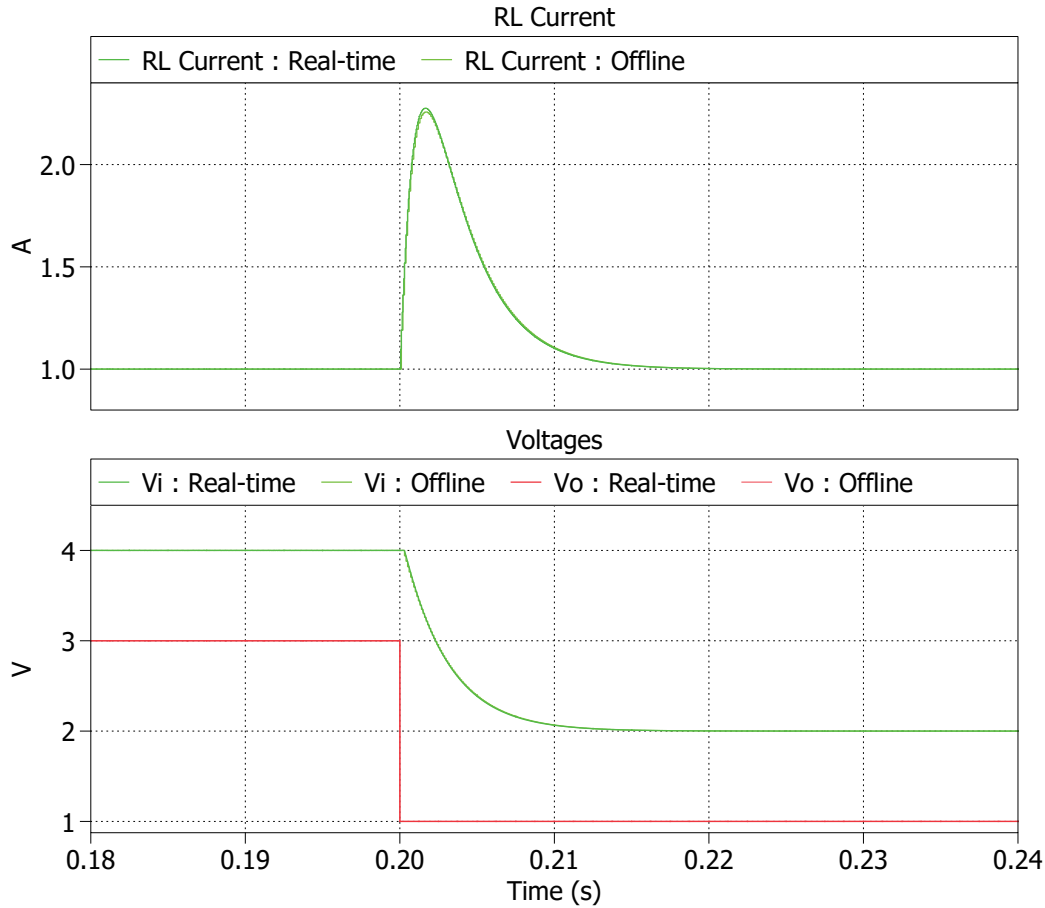


図4: オフラインおよびリアルタイムの結果のベンチマーク - 過渡現象の拡大表示



4 まとめ

このモデルは、シンプルなRL回路とPIコントローラを使用して、NI VeriStand Target Support Packageで仮想プロトタイピングシステムを作成する方法を示しています。仮想プロトタイピング構成では、プラントとコントローラの両方がターゲットハードウェア上で仮想的に実行されます。ハードウェアI/Oはループバックされ、システムの閉ループ動作を可能にします。

5 参考文献

[1] NI, *VeriStand Manual*, 2020,

URL: <https://www.ni.com/documentation/en/veristand/latest>.

[2] Plexim, *NI VeriStand Target Support User Manual*,

URL: <https://adv-auto.co.jp/products/plexim/manual.html>

改訂履歴:

NI VeriStand TSP 1.0.1 初版

Pleximへの連絡方法:

☎ +41 44 533 51 00	Phone
✉ Plexim GmbH	Mail
Technoparkstrasse 1	
8005 Zurich	
Switzerland	
@ info@plexim.com	Email
https://www.plexim.com	Web

計測エンジニアリングシステムへの連絡方法:

☎ +81 3 6273 7505	Phone
✉ Keisoku Engineering System CO.,LTD.	Mail
1-9-5 Uchikanda, Chiyoda-ku	
Tokyo, 101-0047	
Japan	
https://kesco.co.jp	Web

NI VeriStand TSP Demo Model

© 2002–2022 by Plexim GmbH

このマニュアルで説明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の書面による事前の同意なしに、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。

本マニュアルは、Plexim社の英文マニュアルを日本語に翻訳したものです。本マニュアルと英文マニュアルとで差異がある場合、英文マニュアルを正とします。

本マニュアルの内容に基づいて発生した負傷や損害などに対して、Plexim GmbHおよび計測エンジニアリングシステム株式会社は一切責任を負いません。製品とアプリケーションに関連したリスクを最小限に抑えるため、ユーザが適切な設計および保護対策を用意する必要があります。