

# Embedded Code Generation *DEMO MODEL*

## SVPWM Control of a Grid-Connected Three-Level NPC Inverter

グリッド接続型3レベルNPCインバータのSVPWM制御

- TI C2000 MCU用組み込みコード生成によるNPCインバータの制御 -

Last updated in C2000 TSP 1.9.1

# 1 はじめに

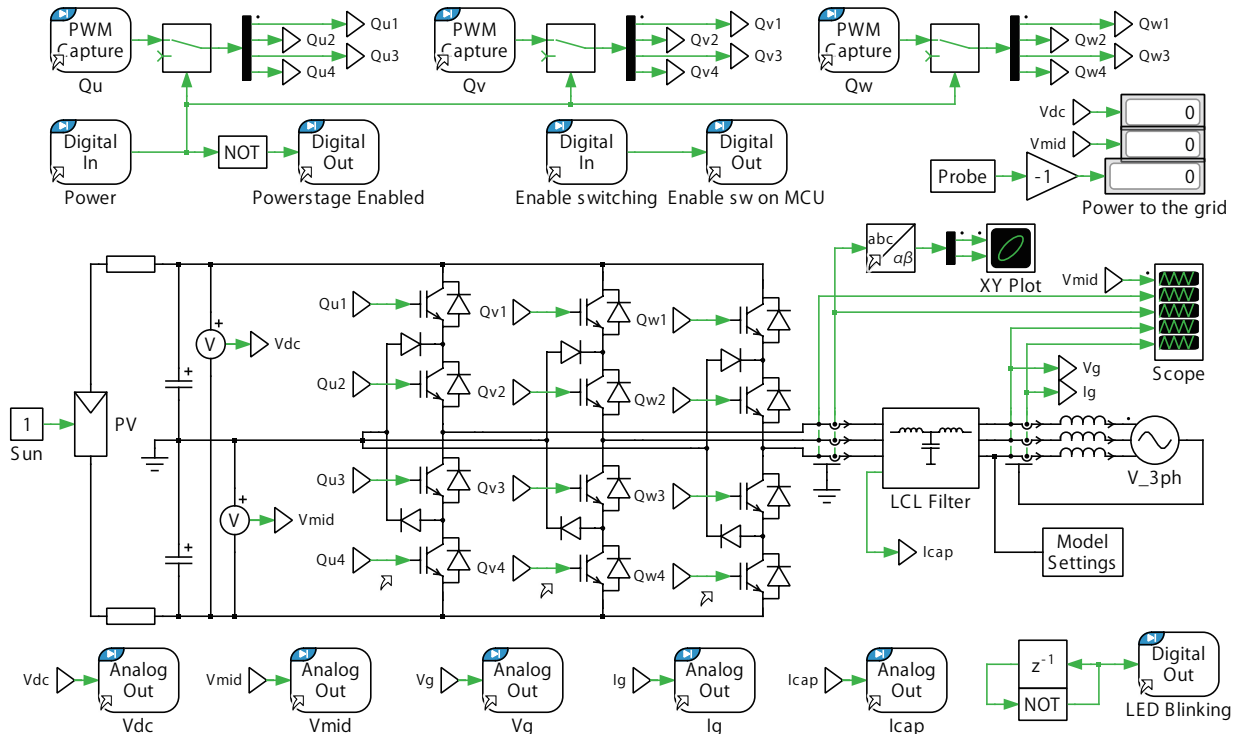
このデモ モデルは、空間ベクトルPWM(Space-Vector PWM: SVPWM) と中性点電位バランス技術を使用した閉電流ループ内のグリッド接続NPCインバータのシミュレーションを示します。Texas Instruments (TI) C2000 MCUを使用したPLECS Embedded Coder の一般的なワークフローについて説明します。PLECS RT Boxと組み合わせることで、MCUの性能をリアルタイムに検証できます。

**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\*

図1: グリッド接続NPCインバータシステムの電源回路



## 2 モデル

デモモデルは2つのサブシステムで構成しています。"Plant"サブシステムには電源回路が含まれ、"Controller"サブシステムには制御ループとSVPWM変調方式が含まれています。

次の2.1と2.2では、電源回路と制御について詳しく説明しています。必要に応じて第3章に進み、リアルタイムシミュレーションの実行手順を確認してください。

### 2.1 電源回路

電源回路には、LCLフィルタを介してグリッドに接続された3相の中性点クランプ(Neutral-Point Clamped: NPC)インバータが含まれます。DC入力は、"Sun"が公称放射輝度レベル(つまり値1)にあるときに800Vのフル電圧を供給します。2つのDCキャパシタは、インバータの上部と下部に個別に入力を供給します。したがって、SVPWMアルゴリズムには中性点電位バランス

技術が組み込まれています。3レベルの電圧源インバータ(Voltage Source Inverter: VSI)は、3つのIGBT 3-レベルハーフブリッジ(NPC)パワー素子モジュールを使用して表現します。

12個のPWMスイッチング信号は、PLECS RT Boxコンポーネントライブラリの3つのPWM Captureブロックを使用してサブシステムに取り込まれます。DC電圧、AC電流、AC電圧、およびフィルタキャパシタ電流の測定値は、Analog Outポートを介してサブシステムから出力されます。"Enable sw on MCU"というラベルのDigital Outブロックは、外部デジタル入力信号をRT Boxのデジタル出力ピンに転送します。このピンはMCUのGPIOに接続され、制御ソフトウェアに実装された有限ステートマシンを介してPWM出力を有効/無効にします。MCUのPWM有効/無効機能のメカニズムについては、[2.2](#)で詳しく説明します。

### LCLフィルタの設計

3レベルインバータの出力ノードでの脈動電圧のため、スイッチング高調波を減衰するには、インバータとグリッド間に何らかのフィルタを挿入する必要があります。フィルタは通常、インダクタとキャパシタで構成されており、インバータ制御によってインバータがグリッドと、有効電力および無効電力とを交換できるようにします。ここで使用するLCLフィルタの設計のガイドラインは参考文献[1]に基づいています。

**フィルタキャパシタ $C_f$ の設計**  $C_f$ を設計するには、LCLフィルタによる無効電力の変動を考慮する必要があります。設計の基本値は次のように定義できます:

$$Z_b = \frac{V_{LL}^2}{P_n}, C_b = \frac{1}{\omega_g Z_b}, L_b = \frac{Z_b}{\omega_g}$$

ここで、 $V_{LL}^2$ はグリッド線間RMS電圧、 $P_n$ は定格有効電力、 $\omega_g = 2\pi f_g$ はグリッド角周波数です。フィルタの静電容量は、基本値 $C_b$ のパーセンテージとして表されます。グリッドで観測される最大電力変動を5%(経験値1% - 5%)に設定すると、次の式が得られます:

$$C_f = 0.05C_b$$

**フィルタインダクタ $L_c$ と $L_g$ の設計** 低い周波数では、LCLフィルタはインダクタのように動作し、次のような総インダクタンスになります:

$$L_{dc} = L_c + L_g$$

フィルタキャパシタ $C_f$ を流れる電流は、フィルタインダクタの電流に比べて無視できるほど小さいことに注意してください。 $L_c$ と $L_g$ 間には、基本周波数における電圧降下を最小限に抑え、フィルタリング能力を最大化する関係があります。 $L_c$ と $L_g$ を $L_{dc}$ に対するパーセンテージとして $\alpha \in [0, 1]$ で表す場合、次のようになります:

$$L_c = \alpha L_{dc}, L_g = (1 - \alpha)L_{dc}$$

LCL共振回路は $L_c$ 、 $C_f$ 、 $L_g$ がすべて並列に接続されているため、LCL共振周波数は次のようになります:

$$\omega_r = \sqrt{\frac{L_c + L_g}{L_c C_f L_g}}$$

したがって、共振周波数を設定する等価インダクタンス $L_{eq}$ は、 $L_c$ と $L_g$ の並列接続として定義できます:

$$L_{eq} = \frac{L_c L_g}{L_c + L_g} = \alpha(1 - \alpha)L_{dc}$$

最小 $L_{eq}$ は、 $L_c = L_g = 0.5L_{dc}$ のときに達成します。 $L_{dc}$ を基本値 $L_b$ の10%(経験値)に設定することで、 $L_c$ と $L_g$ は容易に決定します:

$$L_c = L_g = 0.05L_b$$

フィルタキャパシタ電流を用いたアクティブダンピング技術

LCL共振周波数における高いピークゲインを抑制するために、入力フィルタ用の減衰抵抗を配置することでパッシブダンピングを実現できます[2]。パッシブダンピングは単純な手法ですが、さらなる損失が発生します。一方、アクティブダンピングは、仮想抵抗器による制御の変更を通じて問題を解決し、損失を伴いません。

図2は単相等価理想LCLフィルタとそれに対応するブロックダイアグラムを示しています[1]。図3は減衰抵抗がフィルタキャパシタと直列に配置された減衰フィルタの場合を示しています[1]。図3のブロックダイアグラムは、仮想ゲイン成分(図3の"RD"をエミュレート)を使用することで、物理的なダンピング抵抗の電力損失を除去しながら、理想LCLフィルタの制御方式と同等のダンピング効果を追加できることを示しています。アクティブダンピング手法では、フィルタキャパシタ電流を測定する必要があり、これには追加で3つ(バランスの取れた三相電流の場合は2つ)のアナログ入力/出力が必要になります。図2のフィルタは、sドメインの状態方程式で記述できます：

図2: 単相等価理想LCLフィルタと対応するブロックダイアグラム

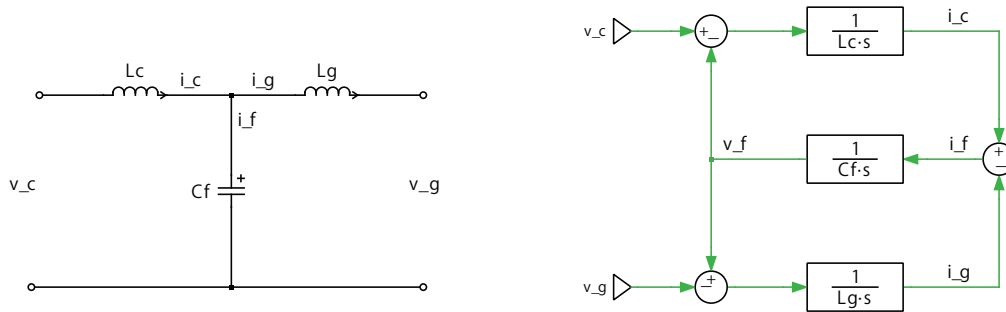
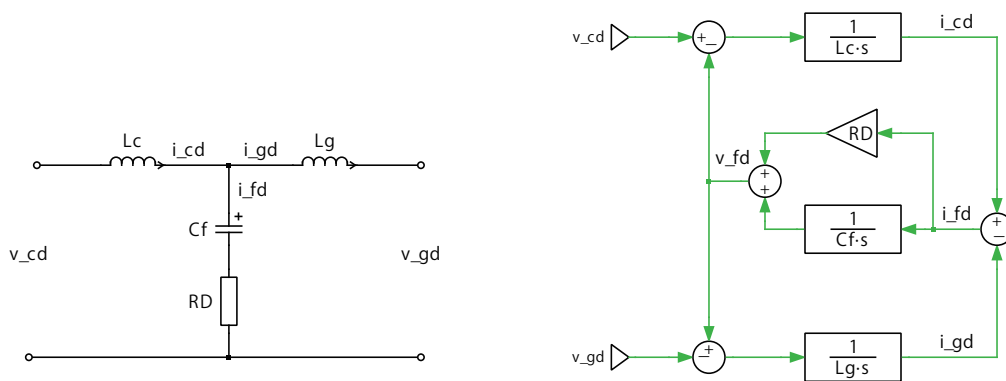


図3: 単相等価減衰LCLフィルタとモデル、キャパシタと直列の減衰抵抗



$$v_c(s) = sL_{cic}(s) + v_f(s)$$

$$v_g(s) = v_f(s) - sL_g i_g(s)$$

$$v_f(s) = \frac{i_f(s)}{sC_f}$$

$$i_c(s) = i_f(s) + i_g(s)$$

したがって、理想LCLフィルタの電圧と電流の伝達関数は次のようになります：

$$H_{v_f}(s) = \frac{v_f(s)}{v_c(s)} = \frac{1}{L_c C_f s^2 + \omega_f^2}$$

$$H_{i_f}(s) = \frac{i_f(s)}{v_c(s)} = \frac{i_f(s)}{v_f(s)} \frac{v_f(s)}{v_c(s)} = \frac{1}{L_c} \frac{s}{s^2 + \omega_f^2}$$

ここで $\omega_r$ は前に定義した共振周波数です。図3の減衰抵抗付きLCLフィルタの場合、次の式が得られます:

$$v_c^d(s) = s \cdot L_c \cdot i_c^d(s) + v_f^d(s)$$

$$v_g^d(s) = v_f^d(s) - s \cdot L_g \cdot i_g^d(s)$$

$$v_f^d(s) = i_f^d(s) \left( R_D + \frac{1}{sC_f} \right)$$

$$i_c^d(s) = i_f^d(s) + i_g^d(s)$$

ダンピング抵抗 $R_D$ は、共振周波数におけるフィルキャパシタ $C_f$ のインピーダンスの3分の1として選択されることが多くあります[1]:

$$R_D = \frac{1}{3\omega_r C_f}$$

したがって、同じ関心のある電圧と電流の伝達関数を導き出すことができます:

$$H_{v_f}^d(s) = \frac{v_f^d(s)}{v_c^d(s)} = \frac{1}{L_c C_f} \frac{1 + sR_D C_f}{s^2 + \omega_r^2(1 + sR_D C_f)}$$

$$H_{i_f}^d(s) = \frac{i_f^d(s)}{v_c^d(s)} = \frac{i_f^d(s) v_f^d(s)}{v_f^d(s) v_c^d(s)} = \frac{1}{L_c} \frac{s}{s^2 + \omega_r^2(1 + sR_D C_f)}$$

減衰抵抗がない場合には、次のように定義します:

$$H_{v_f}(s) = \frac{v_f(s)}{v_c(s)} = \frac{L_g}{s^2 L_c C_f L_g + L_c + L_g} = H_1(s)$$

$$\frac{i_f(s)}{v_f(s)} = sC_f = H_2(s)$$

減衰抵抗がある場合には次のように定義します:

$$H_{v_f}^d(s) = \frac{v_f^d(s)}{v_c^d(s)} = \frac{L_g(1 + sR_D C_f)}{s^2 L_c C_f L_g + (L_c + L_g)(1 + sR_D C_f)} = H_3(s)$$

アクティブダンピング技術では、物理的な $R_D$ を使用せずに、抵抗値 $R_D$ の仮想抵抗器を使用して $H_3(s)$ を実現したいと考えています。図4は、 $K_{AD}$ ゲインを備えたアクティブダンピング制御ループのブロック図を示しています。図4の閉ループ伝達関数が $H_3(s)$ に類似するようにします。

図4: キャパシタ電圧/電流をフィードバックとして用いたアクティブダンピング制御ループのブロックダイアグラム

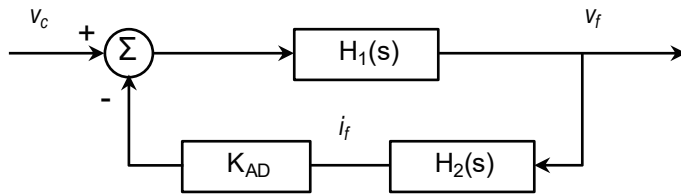


図4では、

$$H_{CL}(s) = \frac{H_1(s)}{1 + H_1(s)H_2(s)K_{AD}} = \frac{\frac{L_g}{s^2 L_c C_f L_g + L_c + L_g}}{1 + \frac{L_g}{s^2 L_c C_f L_g + L_c + L_g} sC_f K_{AD}} = \frac{L_g}{s^2 L_c C_f L_g + L_c + L_g + sL_g C_f K_{AD}}$$

$K_{AD}$ を次のように選択します:

$$K_{AD} = \frac{L_c + L_g}{L_g} R_D$$

以下が得られます:

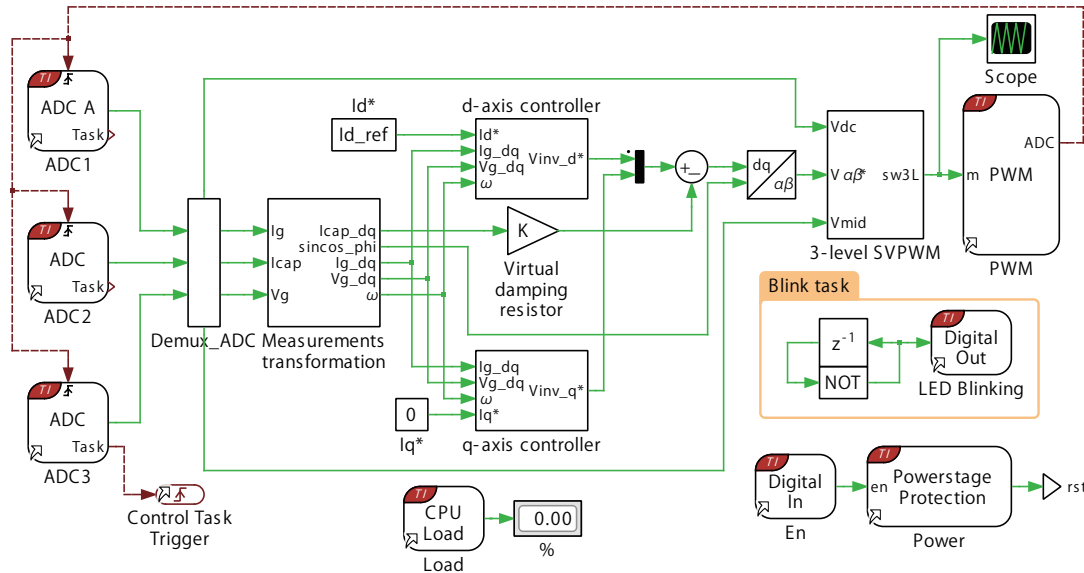
$$H_{CL}(s) = \frac{L_g}{s^2 L_c C_f L_g + L_c + L_g + s L_g C_f \frac{L_c + L_g}{L_g} R_D} = \frac{L_g}{s^2 L_c C_f L_g + (L_c + L_g)(1 + s R_D C_f)}$$

この最終的な $H_{CL}(s)$ は $H_3(s)$ に十分近いので、 $H_3(s)$ の分子の項" $R_D C_f$ "は1よりはるかに小さいため無視できます。実装は、 $i_d$ と $i_q$ の両軸レギュレータについて考慮する必要があることに注意してください。この仮想減衰ゲイン $K_{AD}$ は、コントローラサブシステム(図5を参照)内のゲイン(利得)ブロック"Virtual damping resistor"として配置されています。

## 2.2 Controller

"Controller"サブシステムには、2つの閉ループd-q電流コントローラと、中性点電位バランス調整方式を備えた3レベルSVPWMが実装されています。図5はコントローラモデルを示しています。TI C2000 TargetコンポーネントライブラリのADCブロックとPWMブロックが含まれています。DCリンク電圧、AC電流、AC電圧、およびフィルタキャパシタ電流の測定は、ADCブロックを使用してモデル環境に導入されます。検出したアナログ電圧を制御アルゴリズムで使用する物理単位の値に変換するため、ADCブロックのパラメータウィンドウから各チャンネルにスケーリング係数とオフセットを提供します。ADC unitとAnalog input channelパラメータは、さまざまなMCUの使用可能なリソースに応じて変更できます。

図5: グリッド接続NPCインバータシステムのControllerモデル



$i_d$ および $i_q$ 電流コントローラは、次の伝達関数に基づいて設計されます。

$$H_{OL}(s) \approx \frac{1}{R_g + R_c + (L_g + L_c)s}$$

これにより、フィルタキャパシタに流入する微小電流と、スティッフなグリッドインダクタンス $L_{grid} \ll L_g$ が無視されます。

### SVPWM方式

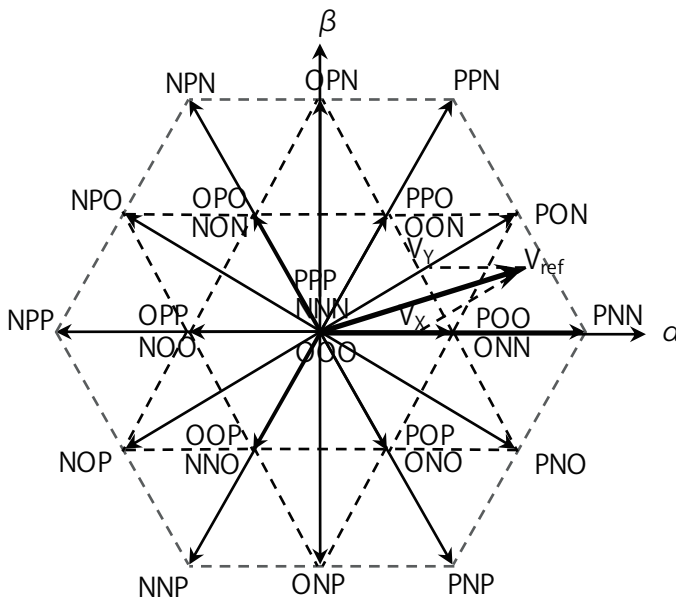
図1には3つのNPCレッグ(位相 u、v、w)が示されています。各レッグには4つのスイッチ $Q_{x1}$ 、 $Q_{x2}$ 、 $Q_{x3}$ 、 $Q_{x4}$ ( $x = u, v, w$ )が含まれており、これらの4つのスイッチは2つの補完的なペアで制御する必要があります。 $Q_{x1}$ と $Q_{x3}$ で1つの補完ペアを形成し、 $Q_{x2}$ と $Q_{x4}$ でペアを形成します。これら4つのスイッチを制御することにより、インバータの出力は3つの異なる電圧レベルを可能にします。表1には、各レッグに対して有効な3つの状態がリストされています。ここで、Pはスイッチレッグが正のDCレールに接続されていること、Nは負のDCレールに接続されていること、Oは中性点電位に接続されていることを意味します。

表1: x1-Qx4の異なるスイッチングの組み合わせに基づく各レッグの出力ステータス

| Switch No. | Qx1 | Qx2 | Qx3 | Qx4 | Phase voltage       | Leg status |
|------------|-----|-----|-----|-----|---------------------|------------|
| 1          | ON  | ON  | OFF | OFF | $\frac{V_{dc}}{2}$  | P          |
| 2          | OFF | OFF | ON  | ON  | $-\frac{V_{dc}}{2}$ | N          |
| 3          | OFF | ON  | ON  | OFF | 0                   | O          |

したがって、図6に示す空間ベクトル図にマッピングできる3レベルVSIの状態は合計27個あります。リファレンスベクトル $V_{ref}$ を仮定すると、SVPWMの理論によれば、 $V_{ref}$ を合成するためには、2つの最も近いベクトル $V_x$ 、 $V_y$ と1つのゼロベクトル $V_z$ を見つける必要があります。したがって、図6のベクトルPNN( $V_x$ )、PON( $V_y$ )、およびNNN( $V_z$ )は、それに応じて選択して、 $V_{ref}$ を形成することができます。

図6: 三相3レベルインバータSVPWMのベクトル図



スイッチング周期 $T_{sw}$ 内のベクトル $V_x$ 、 $V_y$ 、 $V_z$ のドウェル時間がそれぞれ $T_x$ 、 $T_y$ 、 $T_z$ の場合、次の関数を満たす必要があります:

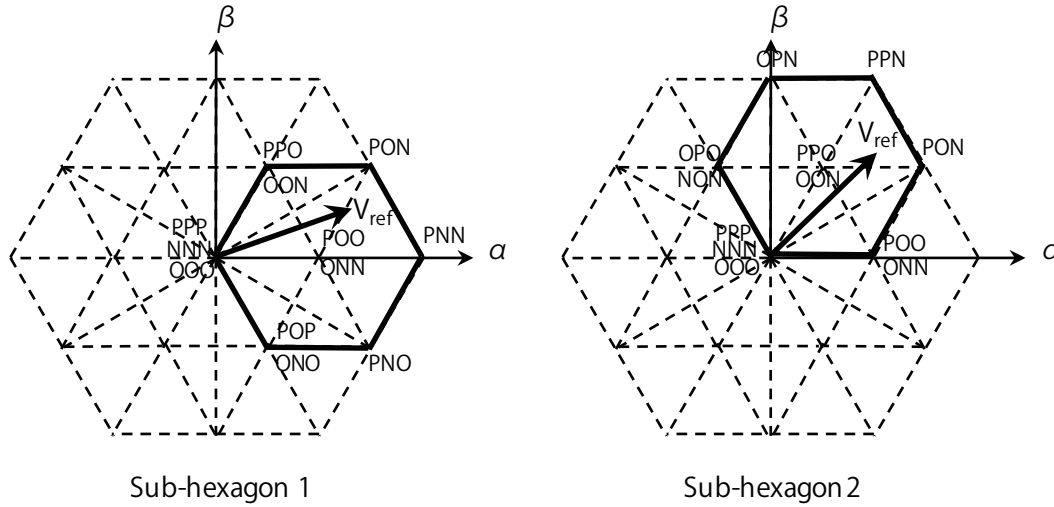
$$V_x T_x + V_y T_y + V_z T_z = V_{ref} T_{sw} \tag{1}$$

$$T_x + T_y + T_z = T_{sw} \tag{2}$$

しかし、2レベルSVPWM方式で使用される角度のみで $V_x$ 、 $V_y$ 、および $V_z$ を決定することは困難です。角度が同じであっても、リファレンスベクトルが異なるセクタに配置される可能性があるためです。セクタを決定するには、リファレンスベクトルの振幅も必要ですが、これにより計算の複雑さが増します。

そこで[3]は2レベルSVPWMのコアを使用して $V_x$ 、 $V_y$ 、 $V_z$ を決定する簡略化された方法を提示しました。まず、図6に示す全体のベクトル図は6つの主要セクタに分割されます。各主要セクタはサブ六角形の形状をしており、6つのサブ六角形はすべて60度の角度差で連続的に分布しています。図7は、サブ六角形1と2を例として示しています。

図7: 全体のベクトル図における6つに分割したサブ六角形のうちのサブ六角形1と2

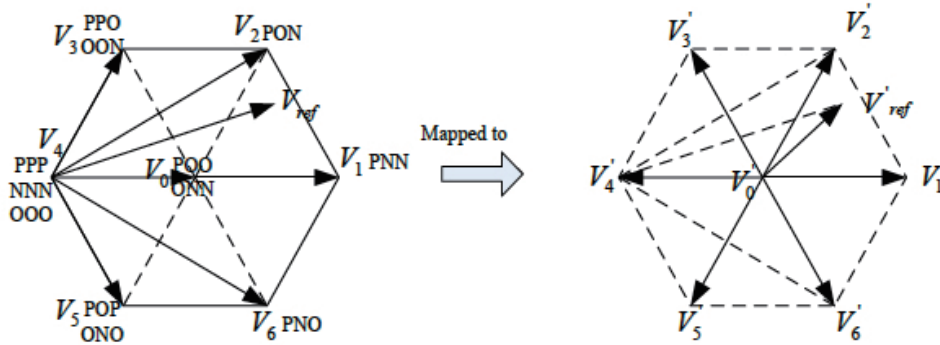


メインセクタ(サブ六角形)を決定した後、元のベクトルを選択したメインセクタにマッピングする必要があります。マッピングアルゴリズムは以下のとおりです:

$$V' = V_{\text{original}} - V_{\text{map}} \tag{3}$$

たとえば、メインセクタ1の元のベクトルは、PPP(OOO, NNN)、POP(ONO)、PNO、PNN、PON、PPO(OON)、POO(ONN)です。2レベルSVPWMに同様の六角形を取得するには、POO(ONN)をマッピングベクトル $V_{\text{map}1} = V_0$ として使用します。マッピング後、[図8](#)に示す六角形が得られます。これは、2レベルSVPWMと同じベクトルダイアグラムです。

図8: サブ六角形1における3レベルSVPWMから2レベルSVPWMへのマッピング



[図8](#)から、 $V_{\text{ref}}$ は依然としてマッピングされたサブ六角形1内にあることがわかり、ドウェルのベクトルが $V_1$ と $V_2$ であることが容易に判断できます。 $V_0$ は、2レベルSVPWMにおけるゼロベクトルとして捉えることができます。したがって、次の関数が得られます。

$$V_1 T_X + V_2 T_Y + V_0 T_Z = V_{\text{ref}} T_{\text{sw}} \tag{4}$$

ここでも[式\(2\)](#)は依然として有効であることに注意してください。[式\(2\)](#)と[式\(4\)](#)を組み合わせると、次の式が得られます:

$$(V_1 - V_{\text{map}1}) T_X + (V_2 - V_{\text{map}1}) T_Y + (V_0 - V_{\text{map}1}) T_Z = (V_{\text{ref}} - V_{\text{map}1}) T_{\text{sw}} \tag{5}$$

すなわち:

$$V_1 T_X + V_2 T_Y + V_0 T_Z = V_{\text{ref}} T_{\text{sw}} \tag{6}$$

したがって、 $V'_1, V'_2, V'_0$ のドウェル時間を計算できれば、元のベクトルのドウェル時間を決定できます。図8のマッピングから、3レベルSVPWMのベクトル選択とドウェル時間計算は2レベルSVPWMに完全に変換されます。

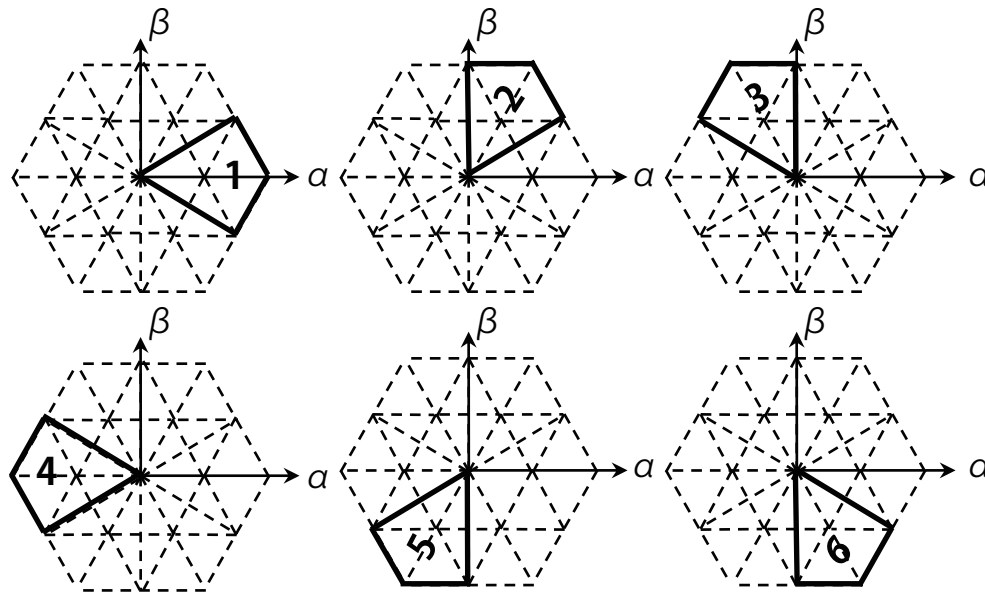
主要セクタが異なれば、マッピングベクトルも異なります。表2は、各主要セクタのマッピングベクトルをまとめたものです。

表2: 各主要セクタのマッピングベクトル

| Main sector No. | Mapping vector | Element of $\alpha$ | Element of $\beta$          |
|-----------------|----------------|---------------------|-----------------------------|
| 1               | POO or ONN     | $\frac{V_{dc}}{3}$  | 0                           |
| 2               | PPO or OON     | $\frac{V_{dc}}{6}$  | $\frac{\sqrt{3}V_{dc}}{6}$  |
| 3               | OPO or NON     | $-\frac{V_{dc}}{6}$ | $\frac{\sqrt{3}V_{dc}}{6}$  |
| 4               | OPP or NOO     | $-\frac{V_{dc}}{3}$ | 0                           |
| 5               | OOP or NNO     | $-\frac{V_{dc}}{6}$ | $-\frac{\sqrt{3}V_{dc}}{6}$ |
| 6               | POP or ONO     | $\frac{V_{dc}}{6}$  | $-\frac{\sqrt{3}V_{dc}}{6}$ |

メインセクタ番号は、 $\alpha$ - $\beta$ 座標平面における $V_{ref}$ の角度によって定義できます。例えば、図7では、メインセクタ1の角度範囲は $[-\frac{\pi}{3}, \frac{\pi}{3}]$ であり、メインセクタ2の角度範囲は $[0, \frac{2\pi}{3}]$ です。したがって、メインセクタ1と2間の重複領域は、各セクタの角度領域を独占するために、2つの隣接する領域に均等に分割できます。図9は、6つの主要セクタの簡略化した定義を示しています。

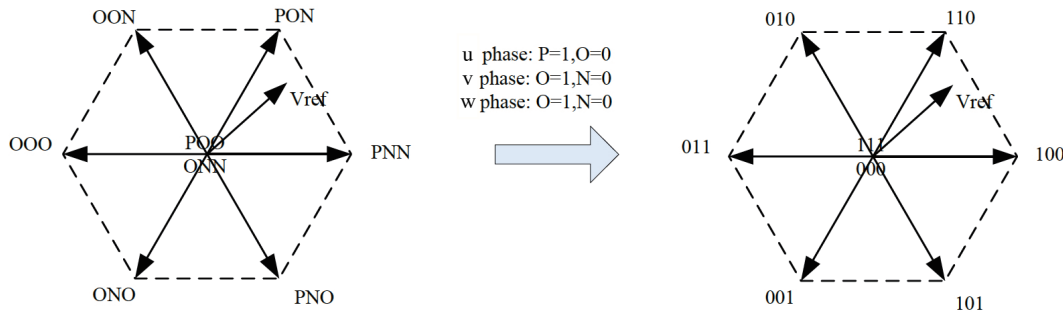
図9: 6つの主要セクタの簡略化した定義



2レベルSVPWMでは、最初のステップは、ドウェルのベクトルを決定できるセクタ番号を見つけることです。2番目のステップは、選択したベクトルごとにドウェル時間を計算することです。3レベルSVPWMの原理によれば、メインセクタを決定し、すべてのベクトルがメインセクタにマッピングされた後、2レベルSVPWMで行われたのと同じプロセスを適用してサブセクタを決定し、各ドウェルのベクトルのドウェル時間を計算できます。このデモモデルでは、各ドウェルのベクトルのドウェル時間から各スイッチングペアのデューティ比を計算するためのシンプルで効果的な方法[3]を採用しています。

再びメインセクタ1を例に挙げます。図7によれば、u相にはNステータスはありませぬ。また、ベクトルマッピングにOON、ONO、およびOOOを選択した場合、v相とw相にはPステータスがありません。u相では、Pステータスを1で表し、Oステータスを0で表します。v相とw相では、Oステータスを1で、Nステータスを0で表します。図10はこの置換ルールを示しています。このダウンサイジング操作の後、3つのベクトルのドウェル時間を決定できます。図10に示すように、 $T_x$ はステータス100のドウェル時間、 $T_y$ はステータス110のドウェル時間、 $T_z$ はステータス111と000の時間を表します。

図10: 3レベルベクトルダイアグラムを2レベルに縮小する、メインセクタ1のステータス置換ルール



ここでは、対称PWMモードを使用して、3つの補完スイッチペア(d1, d2, d3)のうち上部スイッチの3つのデューティ比を計算できます。図11の左側は、結果として得られる2レベルのベクトルシーケンスが実質的に000 - 100 - 110 - 111 - 110 - 100 - 000であることを示しています。次に、図10と同じステータス置換ルールを適用すると、図11の右側の部分が得られます。したがって、ONN → PNN → PON → POO → PON → PNN → ONNという3レベルのベクトルシーケンスを実現できます。

図11: 対称2レベルSVPWMに状態置換ルールを適用し、3レベルSVPWMに戻す

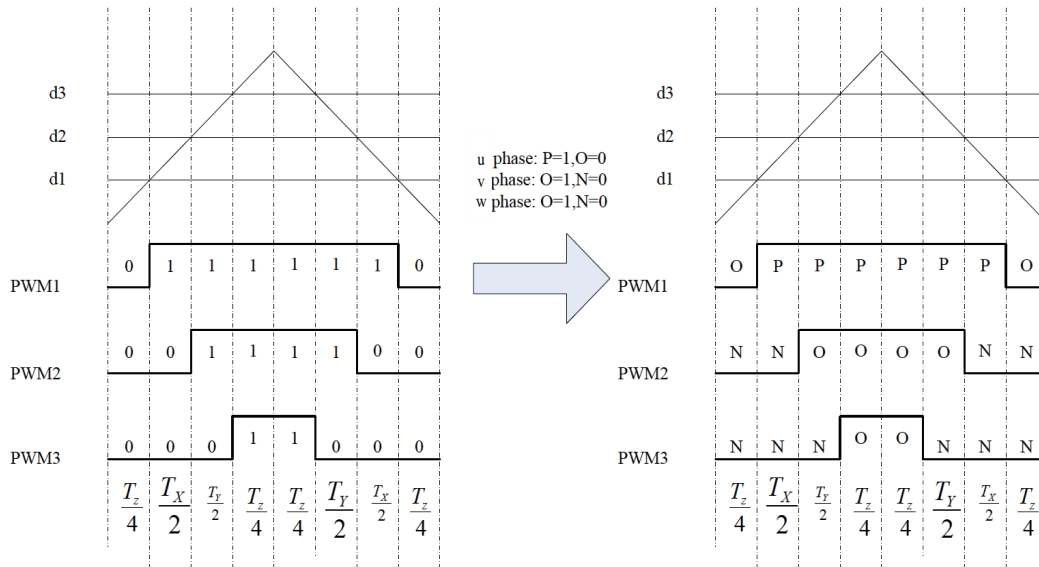


表3は、各主要セクタのステータス置換ルールをまとめたものです。電源スイッチの正のペアは $Q_{x1}$ と $Q_{x3}$ ( $x = u, v, w$ )であり、電源スイッチの負のペアは $Q_{x2}$ と $Q_{x4}$ ( $x = u, v, w$ )です。また、各ペアに対して2レベルSVPWMと同じステータス0と1を定義します。したがって、メインセクタ1の場合、1つのスイッチングサイクルでu相にNステータスがないため、負のペアであるu相は常に0になります。同様に、v相とw相にはPステータスがないため、正のペアは常に0になります。つまり、メインセクタ1では、d1をu相の正のペアに(つまり $Q_{u1}$ のデューティ比はd1で、 $Q_{u3}$ は $Q_{u1}$ と相補的なロジックを持ちます)、d2をv相の負のペアに(つまり $Q_{v2}$ のデューティ比はd2で、 $Q_{v4}$ は $Q_{v2}$ と相補的なロジックを持ちます)、d3をw相の負のペアに割り当てる

ことができます(つまり、Qw2のデューティー比はd3で、Qw4はQw2と相補的なロジックを持ちます)。このプロセスは6つの主要セクタすべてに拡張でき、表4にこのデューティー比の割り当てルールを示します。

表3: 各主要セクタのステータス置換ルール

| Main sector No. | U phase |   | V phase |   | W phase |   |
|-----------------|---------|---|---------|---|---------|---|
|                 | 1       | 0 | 1       | 0 | 1       | 0 |
| 1               | P       | O | O       | N | O       | N |
| 2               | P       | O | P       | O | O       | N |
| 3               | O       | N | P       | O | O       | N |
| 4               | O       | N | P       | O | P       | O |
| 5               | O       | N | O       | N | P       | O |
| 6               | P       | O | O       | N | P       | O |

表4: 各主要セクタのデューティー比割り当てルール

| Main sector No. | U phase             |                     | V phase             |                     | W phase             |                     |
|-----------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|
|                 | Positive pair (Qu1) | Negative pair (Qu2) | Positive pair (Qv1) | Negative pair (Qv2) | Positive pair (Qw1) | Negative pair (Qw2) |
| 1               | d1                  | 1                   | 0                   | d2                  | 0                   | d3                  |
| 2               | d1                  | 1                   | d2                  | 1                   | 0                   | d3                  |
| 3               | 0                   | d1                  | d2                  | 1                   | 0                   | d3                  |
| 4               | 0                   | d1                  | d2                  | 1                   | d3                  | 1                   |
| 5               | 0                   | d1                  | 0                   | d2                  | d3                  | 1                   |
| 6               | d1                  | 1                   | 0                   | d2                  | d3                  | 1                   |

上で説明した簡略化された3レベルSVPWMアルゴリズムは、C言語入力ブロックを使用して実装し、出力は表4にまとめられているように、Qu1、Qu2、Qv1、Qv2、Qw1、Qw2の6つの変調指数値を個別に提供します。それぞれの補完信号は、対応するPWM Outブロックで反対の極性(Polarity)を設定することによって実現します。

### 中性点電位バランス技術

このデモモデルに実装されているアクティブ中性点電位バランス制御は[4]に基づいています。中性点電流NPは、図1に示すDC中性点から出てくる電流として表されます。図6に示すSVPWMベクトル図に基づき、表5は中性点電位のバランスに影響を与えるすべてのベクトルをまとめたものです。u、v、wは、グリッド側に入る3レベル位相レグのスイッチングノードから出る電流の方向を示します。

小型のベクトルは、POOとONNのようにペアで存在し、それらはまったく同じ電流値で中性点を乱しますが、符号は反対です。さらに、図10に示すように、POOとONNは、メインセクタ1の縮小された2レベルベクトルダイアグラムにおける新しいゼロベクトルです。したがって、中性点電圧のアクティブ制御は、この新しいゼロベクトルペアの操作にあります。これには、全DC電圧(図1の $v_{dc}$ )に加えて、中点電圧(図1の $v_{mid}$ )を測定する必要があります。その結果、図11の合計ゼロベクトルのドウェル時間 $T_z$ は、2つのレベル"111"と"000"ベクトルに分割され、それぞれがその半分のDC電圧に比例します。中型のベクトルは中性点を乱し、各メインセクタ内に中型のベクトルとバランスをとるために使用できるベクトルが存在しないため、各メインセクタの

スケールで中性点の変動がいくらか観察されることがあります。この手法の詳細および制限事項については、[\[4\]](#)を参照してください。

表5: 異なる空間ベクトルにおける中性点電流 $i_{NP}$

| Positive Small Vectors | $i_{NP}$ | Negative Small Vectors | $i_{NP}$ | Medium Vectors | $i_{NP}$ |
|------------------------|----------|------------------------|----------|----------------|----------|
| ONN                    | $i_u$    | POO                    | $-i_u$   | PON            | $i_v$    |
| PPO                    | $i_w$    | OON                    | $-i_w$   | OPN            | $i_u$    |
| NON                    | $i_v$    | OPO                    | $-i_v$   | NPO            | $i_w$    |
| OPP                    | $i_u$    | NOO                    | $-i_u$   | NOP            | $i_v$    |
| NNO                    | $i_w$    | OOP                    | $-i_w$   | ONP            | $i_u$    |
| POP                    | $i_v$    | ONO                    | $-i_v$   | PNO            | $i_w$    |

### TI C2000 Targetライブラリコンポーネントの構成

SVPWMモジュレータの出力は、ゲート信号を生成するために [0, 1]の範囲のデューティ比の形式でPWMブロックへの入力として提供されます。PWMブロックパラメータウィンドウの**全般**タブから**Carrier type**と**Carrier frequency**パラメータを、**Dead time**パラメータは**Output**タブから直感的に設定できます。各デューティ比信号は、デッドタイムとは逆の極性で、ePWMモジュールのA出力とB出力の両方に供給されることに注意してください。PWMブロックパラメータ ウィンドウの**Events**タブで、**ADC Trigger**パラメータがUnderflowとして構成しています。これにより、**全般**タブの**PWM generator**で構成された最初のePWMモジュールは、キャリア値が最小値に達するたびに、ADCの"変換開始"インパルスを生成します。この構成では、PWMブロックのマスクに"ADC"という追加の出力ポートが表示されます。

TIマイクロコントローラ28379Dには4つのADCモジュールがあるため、ここではADCモジュールA、B、C を使用して測定値を読み取ります。各ADCブロックでは、**Trigger source**がShow trigger portとして選択されているため、ADCブロックのマスクにトリガのシンボルが表示されます。ADCブロックの各トリガポートは、赤い破線の信号線を介してPWMブロックのADCポートに接続されます。これは、最初のePWMモジュールによって生成された変換開始信号が、ADC A、B、Cの変換を同時にトリガすることを意味します。

制御タスクは、ADC C モジュールでの最後の変換後に実行されます。これは、ADC Cの"Task"出力ポートを、赤い破線の信号線でTI C2000 Targetコンポーネントライブラリの制御タスクトリガブロックに接続することによって構成されます。

実行中にPWM信号を有効または無効にするには、RT Box LaunchPad Interfaceボード上のDIPスイッチ"DI-29"を使用します。"Plant"サブシステムからのこの入力信号"DI-29"は、RT Box LaunchPad Interfaceボードを介して、"Controller"サブシステムのPowerstage Protectionブロックの入力に物理的にルーティングされます。Powerstage Protectionは、安全のために対象デバイス上のすべてのPWM出力を有効または無効にする有限ステートマシンを実装しています。ロジックがLowからHighに遷移するとPWM出力が有効になり、HighからLowに遷移すると無効になります。詳細は、このブロックの**ヘルプ**を参照してください。

Powerstage Protectionブロックへの入力がLowからHighに遷移してPWMが有効になると、Powerstage Protectionブロックの出力はアクティブローに設定されます。この出力は、RT Box LaunchPad Interfaceボードを介して"Plant"サブシステムの"Power"というラベルのDigital Inブロックへの入力として提供され、キャプチャしたPWM信号をインバータブリッジのゲートに渡すことを可能にします。LaunchPad Interfaceボード上の赤色LED "DO-29"が点灯し、視覚的にもそのことが示されます。

### 3 シミュレーション

このデモモデルのシミュレーションは、コンピュータ上でオフライン モードで実行できるだけでなく、"Controller"サブシステムをTI C2000 MCU用のターゲット固有のコードに直接変換することもできます。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 [5]、TI 28P550SJ9 [7]、TI 28P650DK9 [8]、TI 28379D [6] LaunchPad、およびTI 28388D [10] controlCARDをサポートしています。

---

**注意** TI 28P550SJ9 LaunchPadの場合、スイッチS6を"BP"に設定する必要があります。

---

さらに、デモモデルでは、TI 280039C [11]およびTI 28379D controlCARD[9]用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タイプを適切に構成する必要があります。

一般的なハードウェアインザループ(Hardware-In-the-Loop: HIL)構成を図12に示します。ここでは、評価キットであるTI 28379D LaunchPad(赤色のボード)が、RT Box LaunchPad Interface(緑色のボード)を介して RT Box に接続されています。

以下の手順に従って、"Controller"サブシステムをTI MCUにアップロードします。

- USBケーブルで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。
- **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。
- PLECSからMCUを直接Build and programするには、**Build configuration**でFlash or Run from RAMを選択して、MCUフラッシュメモリまたはRAMにプログラムします。そして**Board type**でLaunchPadを選択して、**ビルド**をクリックします。

正しくプログラムされていれば、LaunchPadのLED "D9"が点滅します。

Code Composer Studioに精通している上級ユーザー向けには、Generate code into CCS projectオプションがあります。CCSプロジェクトから適切なcgフォルダを見つけ(手順については[12]を参照)、そのパスを**CCS project directory**フィールドに入力して**ビルド**をクリックします。"Controller"サブシステムのコードが自動的に生成されます。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。

---

**注意** RT Box LaunchPad Interfaceボードを使用する場合は、シミュレーション中は**RST**ジャンパが開いていることを確認してください。

---

プログラムしたMCUで実際のパワーステージを制御する前に、PLECS RT Boxでコントローラの動作を確認し、HILテストを実行することを強くお勧めします。一般的なハードウェア構成を図12に示します。ここでは、評価キットである TI 28379D LaunchPad(赤色のボード)が、RT Box LaunchPad Interface(緑色のボード)を介して RT Box に接続されています。

RT Boxでリアルタイムモデルを実行するには、以下の手順に従います。RT Boxにビルドする前に、適切なTI MCU上に"Controller"サブシステムが構築済みであることを確認してください。

- **Coder** -> **Coderオプション...**ウィンドウの**システム**リストから、"Plant"を選択し、ターゲットRT Boxに**ビルド**します。
- モデルをアップロードしたら、**Coderオプション...**ウィンドウの**外部モード**タブからRT Boxに**接続**し、**自動トリガ**を有効化にチェックして、テスト結果をリアルタイムで観察します。

図12: HILL検証のハードウェア設定



正しくプログラムされていれば、RT Box LaunchPad Interfaceボードの"DO-31"に対応するLEDが点滅します。

RT Box上で実行している"Plant"サブシステムまたは電源回路内では、シミュレートされた電圧と電流が比例してアナログ信号に変換され、RT Boxの前面パネルのAnalog Outコネクタを通じて送信されます。これらのアナログ信号は、RT Box LaunchPad Interfaceボードによってキャプチャされ、TI LaunchPadのADC入力ピンにルーティングされます。次に、MCUはこれらのアナログ信号を処理してPWMスイッチング信号を生成し、Digital Inピンを介してRT Boxに供給します。

2.2の最後で説明しているように、RT Box LaunchPad Interfaceボード上のスイッチ"DI-29"を"Low"から"High"に切り替えて、MCUを有効にします。パワーステージが有効になると、LaunchPad Interfaceの"DO-29"に対応するLEDが点灯します。"Plant"サブシステムのPLECSスコープ内のリアルタイム波形を観察します。

RT Box LaunchPad Interfaceボード上のスイッチ"DI-29"をHighからLowに切り替えると、すべてのゲート信号が無効になります。LaunchPad Interfaceボード上の"DO-29" LEDはオフになります。"DI-29"をHighに戻すと、システムの切り替えが再び有効になります。

---

**注意** この段階で、RT Box LaunchPad Interfaceボード上の"DO-29"に対応するLEDが点灯していることを確認します。

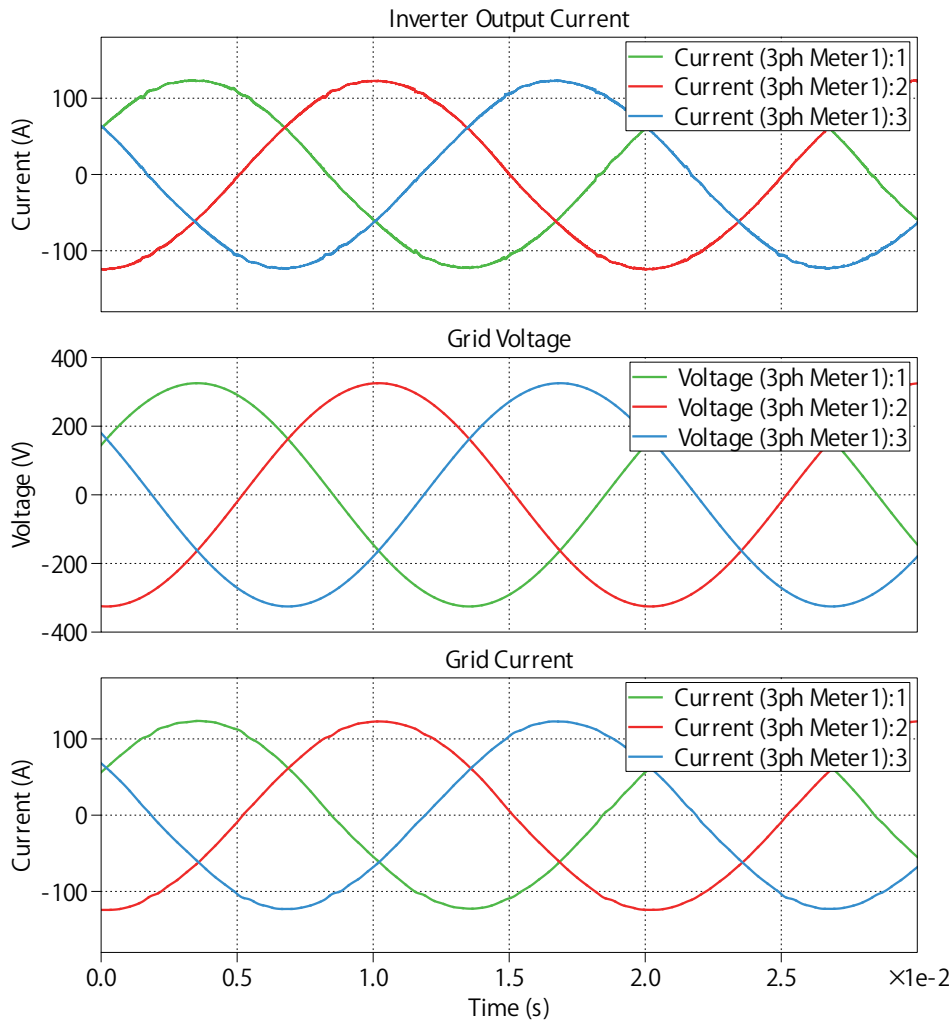
---

MCU内の制御プログラムのパラメータを調整し、中間値を観察するには、以下の手順に従ってTI MCUの外部モードに接続します。

- まず、PLECS RT Boxの**外部モード**から"Plant"サブシステムを切断します(接続している場合)。
- そして、**Coder -> Coder オプション...**ウィンドウ左側の**システム**リストから、"Controller"を選択します。
- 次に、**外部モード**タブから適切な**ターゲットデバイス**を選択し、**接続**をクリックします。
- そして、**自動トリガを有効化**にチェックして、"Controller"サブシステムのPLECSスコープでテスト結果を確認します。
- 次に、RT Boxに**接続**し、**自動トリガを有効化**にチェックして、"Plant"サブシステムのPLECSスコープ上でリアルタイムの結果を観察します。

このデモモデルでは、"Controller"サブシステムの定数ブロック"Id\*"の値を公称値の1.2倍に変更することで、d軸電流リファレンスのステップ変化を構成できます。この値は、**Coderオプション**ウィンドウの**パラメーターのインライン化**タブの**例外**リストに追加されているため、オンザフライで変更できます。このステップ変化の波形は図13で確認できます。

図13: Controllerのd軸電流リファレンスを元の値の1.2倍にリアルタイムで変更した後のインバータ波形

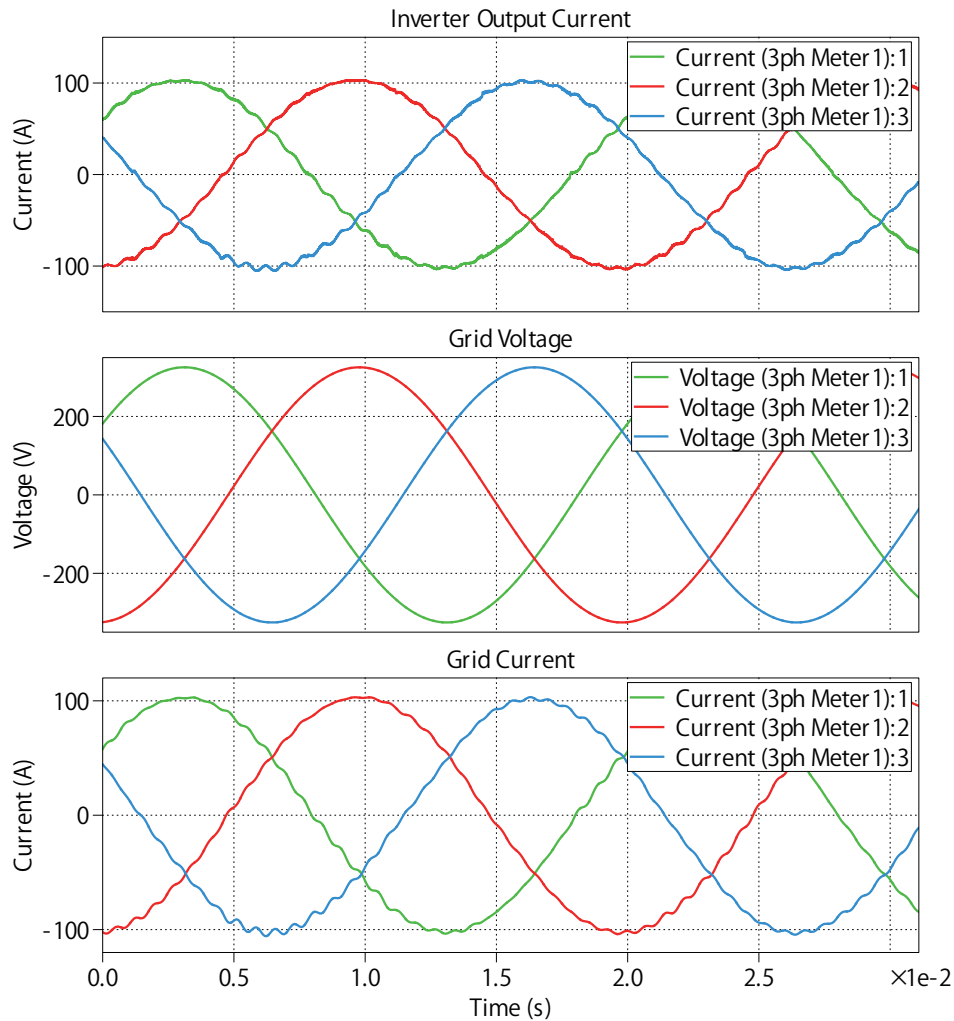


仮想抵抗器がアクティブダンピングに与える影響も観察できます。d軸電流リファレンス $I_d^*$ を変更している場合は、元の値に戻します。次に、"Controller"サブシステムの"Virtual damping resistor"という名前のゲイン(利得)ブロックを元の値の0.7倍に変更します。インバータ電流とグリッド電流の両方で適度なリングングが観察されます。図14はこの現象を示しています。このリングング効果は、LCLフィルタに適用された仮想減衰が不十分なために発生します。"Virtual damping resistor"を元の値に戻すと、リングングがなくなります。

## 4 まとめ

このモデルは、TI C2000 MCUの組み込みコード生成をサポートするグリッド接続型NPCインバータシステムを示しています。

図14: 減衰が不十分な仮想抵抗器によるインバータ波形のリアルタイム表示



## 5 参考文献

- [1] R. Teodorescu, M. Liserre and P. Rodriguez, "Grid converters for photovoltaic and wind power systems", pp. 289-311, IEEE, Wiley, 2011.
- [2] R. W. Erickson, "Optimal single resistors damping of input filters", *APEC '99, Fourteenth Annual Applied Power Electronics Conference and Exposition*, vol.2, pp. 1073-1079, March 1999.
- [3] V. Xue, "Center-Aligned SVPWM Realization for 3-Phase 3-Level Inverter (Application Report)", Texas Instruments, October 2012.
- [4] N. Celanovic and D. Boroyevich, "A comprehensive study of neutral-point voltage balancing problem in three-level neutral-point-clamped voltage source PWM inverters", *IEEE Transactions on Power Electronics*, vol. 15, no. 2, pp. 242-249, March 2000.
- [5] TI C2000 F280039C LaunchPad development kit,  
URL: <http://www.ti.com/tool/LAUNCHXL-F280039C>.
- [6] TI C2000 Delfino MCU F28379D LaunchPad development kit,  
URL: <http://www.ti.com/tool/LAUNCHXL-F28379D>.
- [7] TI C2000 F28P550SJ9 LaunchPad development kit,  
URL: <https://www.ti.com/tool/LAUNCHXL-F28P55X>.

- [8] TI C2000 F28P650DK9 LaunchPad development kit,  
URL: <https://www.ti.com/tool/LAUNCHXL-F28P65X>.
- [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>.
- [11] TI C2000 F280039C controlCARD evaluation module,  
URL: <https://www.ti.com/tool/TMDSCNCD280039C>.
- [12] PLECS TI C2000 Target Support User Manual,  
URL: <https://www.plexim.com/download/documentation>.  
日本語版: <https://adv-auto.co.jp/products/plexim/manual.html>

改訂履歴:

|                 |  |
|-----------------|--|
| C2000 TSP 1.0   | 初版   |
| C2000 TSP 1.5.1 | 28388Dおよび28379Dコントロールカードターゲットのサポートを追加し、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            |
| C2000 TSP 1.9.1 | 28P550SJ9 LaunchPadターゲットのサポートを追加                               |

 Pleximへの連絡方法:

|  |       |
|--|-------|
|  +41 44 533 51 00 | Phone |
| +41 44 533 51 01   | Fax   |
|  Plexim GmbH      | Mail  |
| Technoparkstrasse 1<br>8005 Zurich<br>Switzerland  |       |
| @ info@plexim.com  | Email |
| <a href="http://www.plexim.com">http://www.plexim.com</a>  | 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 |
| +81 3 6285 0250   | Fax   |
|  Keisoku Engineering System CO.,LTD. | Mail  |
| 1-9-5 Uchikanda, Chiyoda-ku<br>Tokyo, 101-0047<br>Japan   |       |
| <a href="https://kesco.co.jp">https://kesco.co.jp</a>   | Web   |

*Embedded Code Generation Demo Model*

© 2002–2024 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。