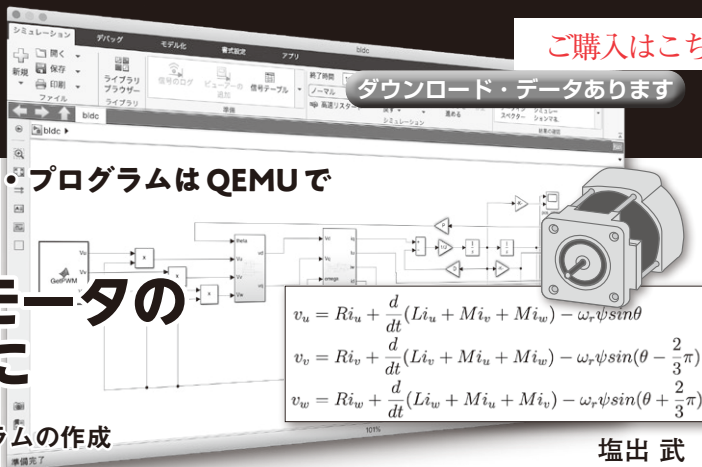


モータ制御はSimulink, マイコン・プログラムはQEMUで

TRY 仮想開発… DCブラシレス・モータの ベクトル制御を例に

第3回 マイコン側モータ制御プログラムの作成

塩出 武



$$v_u = Ri_u + \frac{d}{dt}(Li_u + Mi_u + Mi_w) - \omega_r \psi \sin \theta$$

$$v_v = Ri_v + \frac{d}{dt}(Li_v + Mi_u + Mi_w) - \omega_r \psi \sin(\theta - \frac{2}{3}\pi)$$

$$v_w = Ri_w + \frac{d}{dt}(Li_w + Mi_u + Mi_w) - \omega_r \psi \sin(\theta + \frac{2}{3}\pi)$$

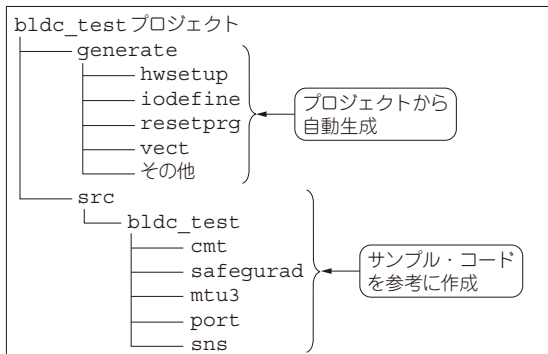


図1 マイコン用制御プログラムの構成

DCブラシレス・モータを回すには高度な制御が必要です。Simulinkを使えばモータ実機なしにシミュレーションで制御が正しいことを確認でき、さらに

QEMUを使えば実機マイコンを使わずに制御を実装したソフトウェアが正しいことを確認できます。

そこで本連載では、DCブラシレス・モータのベクトル制御の基本から、Simulinkによるシミュレーション、QEMUによるソフトウェアのシミュレーションまでを解説します。今回は連載第2回で作成したSimulinkモデルを制御するRXマイコンのC言語のプログラムを作成します。(編集部)

プログラムの構成

マイコン上で実行するベクトル制御用のプログラムを作成します。RX62T搭載低電圧モータ制御評価システム向けのサンプル・プログラムがルネサスエレクトロニクスより公開されているので、今回のシミュレーション評価用にファイル構成や制御割り込みに変

表1 プロジェクトの主なファイル構成

ファイル構成	役割	サンプリング周期	備考
bldc_test	main関数およびUI制御	なし	-
mtu3	ベクトル制御およびエンコーダ監視(cmt側)速度制御も併合	100 μs	速度制御は1ms(10回に1回実施)
cmt	速度制御と始動制御(のライブラリ)	なし	1msの速度制御はmtu3へ併合
port	ポート設定	なし	-
sns	A-Dセンサ監視, LED, SWポート制御	なし	-
safeguard	エラー監視	なし	-
hwsetup	CPUレジスタ初期設定	なし	電源ONリセットで実行

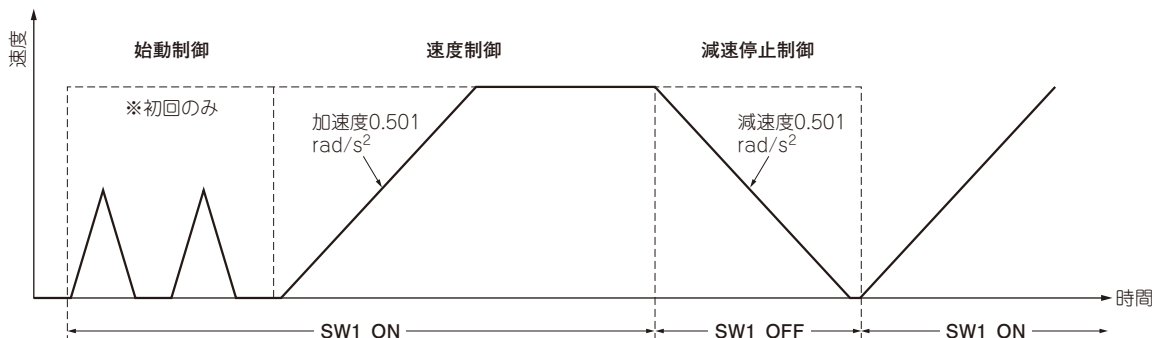


図2 モータ動作概要

