



低ロスと高性能を両立！ きめ細かく制御できる  
ソフトウェア電源を作る

## インバータ/デジタル電源用の定番 マイコン dsPIC33Fプログラミング入門

②クロックとPWMを最高速度・最高分解能に設定する

笠原 政史

Masaji Kasahara

● パワーを制御するには1 ns分解能のPWM信号を生成する必要がある

今回は、本誌で開発したデジタル・パワー制御ボード(写真3)を利用して、dsPICマイコン(写真4)内のペリフェラルの一つである高速PWM回路を動かしてみました。高速PWMを利用すると、目にも止まらないほど速くLEDを点滅させることができ、CPUは何もしなくてもよいことも確認しました。

PWM信号は、ソフトウェアでも生成できますが、処理に多くの時間がかかります。LEDの点滅ぐらいなら十分に対応できますが、パワー制御には遅すぎて処理が間に合いません。

▶CPUで生成するソフトウェアPWMは時間分解能が粗すぎて使い物にならない

高速PWMで得られるPWM信号の分解能は1.04 nsです。一方ソフトウェアで生成したPWMの分解能は、CPUがPWM処理だけに専念したとして、40MIPSで1ループ数命令ぐらいは必要なので、時間分解能は100倍(1.04 ns×100)ぐらいに悪くなります。実際には、CPUはPID制御の計算や液晶ディスプレイ表示などのほかの処理もしますから、1処理あたり100 μs程度要します。その整数倍の時間分解能でPWM生成させると、100000倍(1.04 ns×100000 = 0.104s)ぐらい違います。

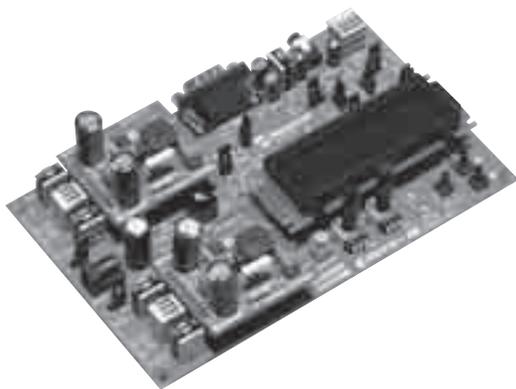


写真3 本誌で開発したパワー制御実験ボード(詳細は本誌2010年10月号特集を参照)

今回は、パワー制御に欠かせないこの「高速PWM」というハードウェアについて理解を深めます。また、今回は手で高速PWMの動作条件を設定しましたが、今回はプログラムで自動設定してみます。

### dsPIC33Fのクロックを 最高速に設定する

● dsPICのクロックと分解能は最高に設定しておく

電源のスイッチング周波数は高くしたほうが、インダクタなどのサイズを小さくできますが、高くしすぎるとPWM設定分解能が相対的に粗くなって制御が安定しません。

このトレードオフを解決する(スイッチング周波数を高くしつつ安定な制御を実現するには、まず、マイコンのクロックをできるだけ高速に設定し、PWM分解能もできるだけ高めておく必要があります。大雑把に言えば、パワー制御用マイコンのクロックとPWMの分解能は、最適値というより最高値に設定しておくわけです。

今回は、そのマイコンの設定法を紹介します。データシートや実験で確認した内容を元に、理解しやすくなるように簡略化した等価回路などで解説します。dsPICは多機能な分、内部のハードウェアが複雑なので、なかなか思いどおり動いてくれません。私の誤解やデータシートの誤記がある場合もありますので、必ず実機で確認しなければなりません。

● 二つのクロック出力

図11に示すのは、dsPIC33Fがもつオシレータ・システムのブロック図です。

図からわかるように、クロック源(源振周波数)を、

写真4 パワー制御実験ボード(写真3)に搭載されているスイッチング電源用dsPICマイコン dsPIC33FJ16GS502

