[第3章]

位置決め制御はメカトロニクスの基本

ステッピング・モータの駆動と位置決め

鶴見 惠一

テーブルやアームを任意の位置に自在に移動させる位置決めはメカトロ技術の基本中の基本です．ステッピング（パルス）モータはこのために存在するようなもので，位置決め装置を最もシンプルな方法で実現できます．

この章では（株）秋月電子通商のステッピング・モータ・ドライブ・キットを利用し，位置制御コントローラを加えて位置制御ができる駆動装置に仕上げます．この章の装置を試みるのにキットは必ずしも必要ではありません．キットを使わない場合はそれに代わる回路例も示しました．

3-1　メカトロで使われるモータとドライブ回路

メカトロと一口に言ってもその幅は広く，大量生産するものから多くても数台しか製作しない産業設備までさまざまです．比較的小型装置の量産品ではモータのドライブ回路から全体の制御ソフトまで設計の対象となるでしょうが，産業設備などではモータのドライブ回路を設計することはまずありません．設計

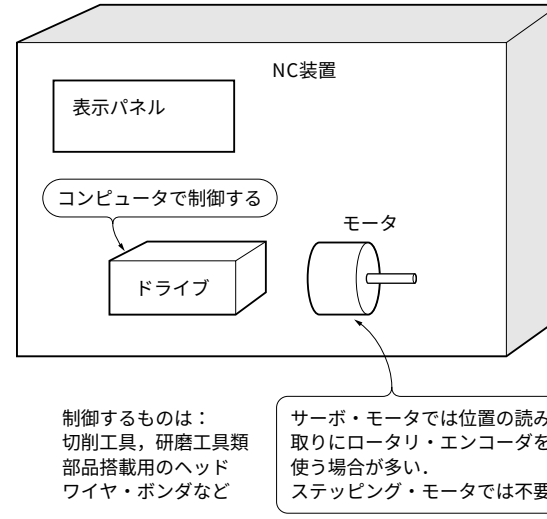

図3-1　産業設備の例
コンピュータで位置制御を行う装置は「NC装置」と呼ばれている．
NCはNumeric Control．数値制御の意味．あらかじめ設定された座標に従って切削・研磨などの自動運転を行う．

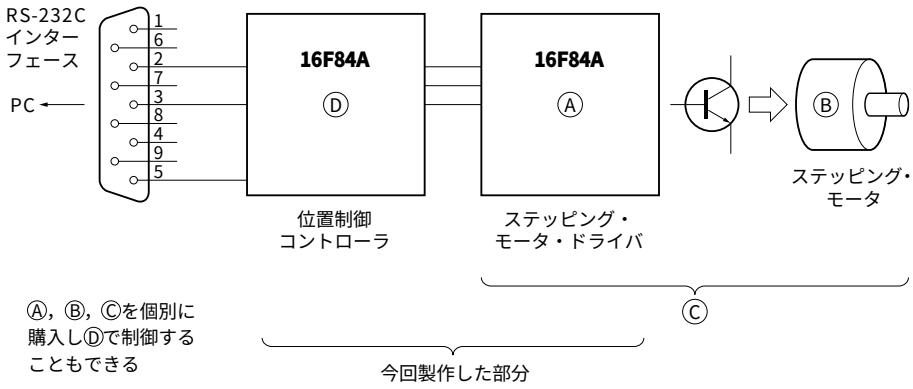

図3-2　構成の略図
位置制御コントローラはステッピング・モータだけでなく，CW/CCW入力のサーボ・モータ・ドライバも駆動できる．Ⓐのドライバはキットを利用したが，ユニバーサル基板で製作も容易．

このアイコンは，章末に用語解説があります

コストがまったく割に合わないのです(**図3-1**参照)．筆者は後者のほうで，仕事でドライブ回路を設計したことはありません．CW/CCWパルスで駆動する市販のドライバを使っています．

ドライバと位置制御コントローラの機能を1個のPIC16F84に組み込むことも可能でしたが，市販のドライバを使った場合の事例としても役立つようにと考え，2個のPIC16F84(または16F84A)で構成しました．CW/CCW駆動に変更されたキットのドライバは一般に市販されているパルス・モータのドライバと同様な感覚で利用できます(**図3-2**参照)．

● アセンブラはマイクロチップ・テクノロジー社版

この章で使うプログラムはすべてマイクロチップ・テクノロジー社版のMPASMWINでアセンブルします．

プログラム開発の効率を考えればCなどの高級言語を利用すべきですが，ハードウェアに密着したプログラムでは，高級言語を使っていてもアセンブリ言語的な思考をしばしば伴う必要があります．つまり，アセンブラでプログラムが作れなければ何も始められない，という状況になりかねないので，この章ではすべてアセンブリ言語で押し通してみました．

3-2　ステッピング・モータ・ドライブ・キットを組み立てる

(株)秋月電子通商のステッピング・モータ・ドライブ・キットです．モータはキットと同時に購入したFDK製のJ28Aを使用しました(Column2参照)．このモータは1回転のステップ数が48で，7.5度/ステップですからかなり荒い精度での位置決めしかできませんが，位置決め装置開発のテクニックを学ぶには十分と思います．早速組み立てて動かしてみましょう(**写真3-1**)．

● 脱調という現象

右回転/左回転に対応したタクト・スイッチを押している間，回転を続けますが，どちらかのスイッチを押したままで回転スピード調整用の*VR*(ボリューム)をゆっくり右に回してみてください．回転速度が

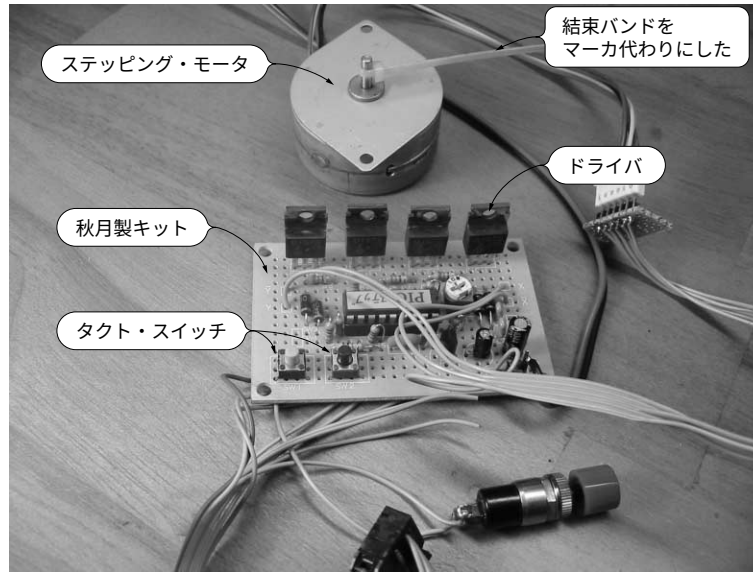

写真3-1　組み立てたステッピング・モータ・ドライブ・キットとモータ
モータ軸に取り付けた結束バンドは停止位置確認用のマーカ．5V電源を使用したのでレギュレータICの78L05は取り外してある．
モータは2相ユニポーラ駆動の小型なものを選ぶ．

徐々に上がり，ある所で急に回転数ががくんと落ち，少々異常な振動が見えると思います．この症状はモータによって少し異なりますが，共通しているのはどんどんスピードを上げていくとある所で急に異常な状態に陥ることです．これをステッピング・モータが脱調したと言います．ステッピング・モータは高速回転が苦手で，明瞭な限界があるのです．

今度は脱調になる手前の*VR*位置を覚えておき，低回転から徐々にスピードを上げて，脱調する手前で*VR*とスイッチから手を放して停止させます．そのままで，もう一度スイッチを押して回転させてください．多分，脱調した状態の異常な振る舞いになると思います．正常に回転可能な範囲であっても，高速で急に回転をスタートできないのです．モータに負荷が加わるとこの現象はさらに顕著になります．

効率良くステッピング・モータを動かすには，低速回転でスタートして徐々にスピードを安全な範囲で上げる，という加速動作が必要です(Column1参照)．

同様に，高速の状態で急に停止させるのも問題です．小型のモータではこの状態を目で確認するのは難しいと思いますが，がくんと停止させることになり，慣性で予定よりも余計に回ってしまい，これも脱調したこととなります．停止する場合は減速動作が必要になるわけです．

加速，高速運転，減速を連続して行う1単位の動作を，速度のグラフの形から台形駆動と呼びます．

3-3　キットをCW/CCW駆動に変更

(株)秋月電子通商のステッピング・モータ・ドライブ・キットのままでは位置決め装置としては使えません．CWまたはCCWのパルス数に正確に反応して，そのステップ数だけモータを動かす装置に変えてみました．

● ハードウェアの変更

ファームウェア に使われているPIC16C56は書き換え可能なフラッシュ・タイプではないので，これを抜き取ってPIC16F84に変えます．ピン配置は同一なので基板はそのままで使えます．スピード調整の回路は不要ですが，邪魔でもないのでそのまま残しておきました．

CW回転/CCW回転だったタクト・スイッチの機能は，CW方向へ1ステップ/CCW方向へ1ステップの機能に変わります．後で作る位置制御コントローラからのCW/CCW駆動信号は，このタクト・スイッチに並列接続します．

● 回路例

キットを使わずに試みる場合は**図3-3**の回路例を参考にしてください．

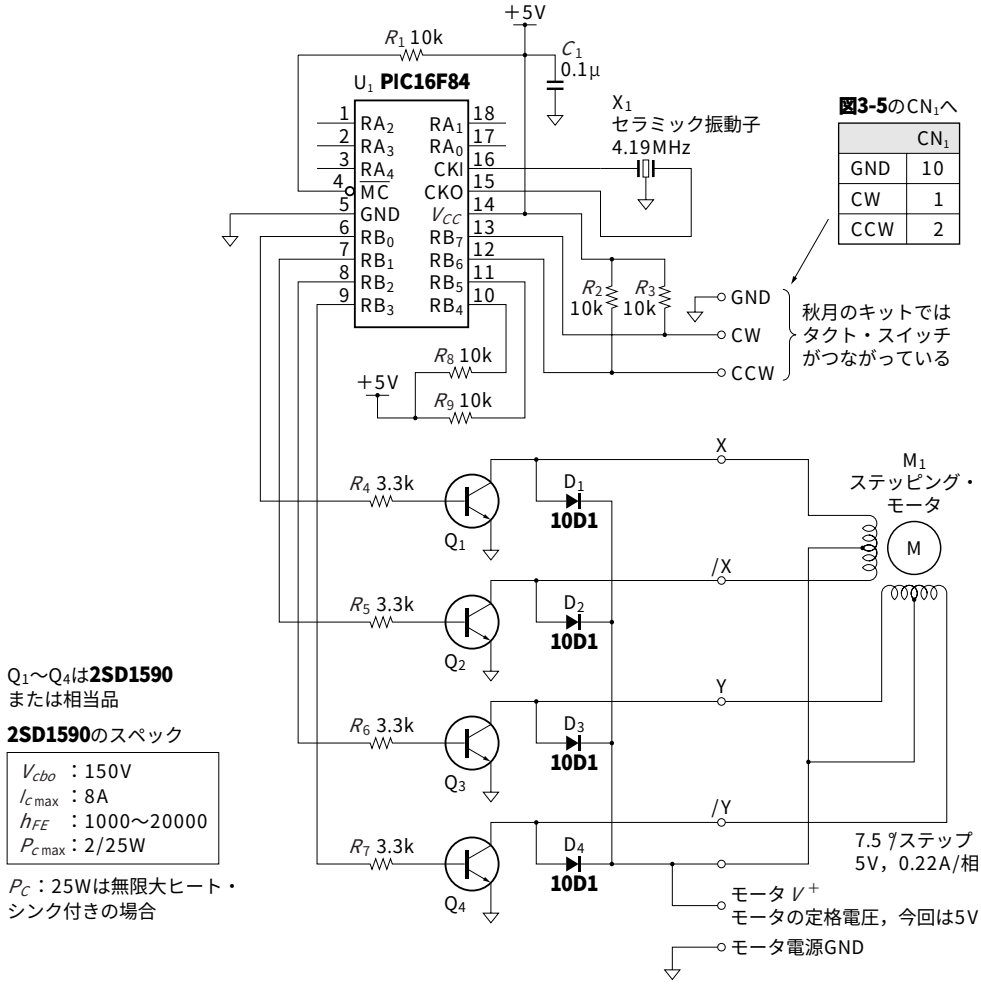

図3-3　ドライブ回路例
(株)秋月電子通商のステッピング・モータ・ドライブ・キットCW/CCW駆動のドライバを試みる場合は，この回路例を参考に．
2相のステッピング・モータを2相励磁，ユニポーラ駆動を行う．

モータのドライブ用のトランジスタ2SD1590(Q_1～Q_4)は内部でダーリントン接続されたものです．2A程度までコレクタ電流が流せるNPNタイプなら何でもかまいませんが，ダーリントンでない場合はベースのシリーズ抵抗(R_4～R_7)は560Ω～1kΩ程度にしたほうが間違いないと思います．

キットに合わせてクロックが4.19MHzになっていますが，重要な意味はありません．3～10MHz程度ならプログラムはそのままで動くはずです．デバッグ・モニタを実行させる場合は，クロックとボー・レートの関係に注意してください．詳細はタイニ・デバッグ・モニタの第5章を参照願います．

● ドライバのプログラムをアセンブル，書き込み，動作確認

ドライバ・プログラムのソースは“PMMagDrv.asm”です．デバッグ・モニタを冒頭でインクルードしているので，“PicMonV2.asm”とそのマクロ定義ファイル“MonMacro.inc”も同じフォルダに置いてアセンブルします．

“PicMonV2.asm”は一部変更が必要です．CW/CCW信号のセンシングには割り込みを使っているので，割り込みベクタと初期化ルーチンのコールを追加します．変更を加えた部分を**リスト3-1**に示しました．

ソースの用意ができたらアセンブルしてPIC16F84に書き込み，動かしてみましょう．

プログラムが起動するとポートと割り込みの初期化を行ってデバッグ・モニタのコマンド待ちになります．この時にRS-232Cインターフェースとパソコンのターミナル・ソフトが用意されていればデバッグ・モニタのコマンドを実行できますが，なくてもかまいません．割り込みは許可されているのでCWまたはCCW入力に反応して以下のように動作します．

(株)秋月電子通商のキットでは右回転/左回転の各タクト・スイッチは，右または左へ1パルスだけ回転させる機能に変わっています．**図3-3**の回路を製作した場合はCWとGND(グラウンド)またはCCWとGNDをつなげると，そのたびに対応した方向へ1パルス分回転します．

リスト3-1　PicMonV2.asm の変更部分
割り込みベクタと初期化ルーチンのコールを追加．網掛けをした部分が追加されている．

```
; Program
        org         0x00
        goto        start
        org         0x04
        goto        RBint                   ;ポートB割り込みサービス
        org         0x05
start
        bsf         STATUS,RP0                      ;banksel  Bank1
        movlw       b'11110'                ;RA.0を出力ポートへ
                                            ;RA.1を入力ポートへ
        movwf       TRISA                   ;banksel  Bank0
        bcf         STATUS,RP0
        bsf         txd

        clrf        statustemp
        clrf        pclathtemp
        clrf        fsrtemp

; ターゲット・プログラム
        call        Init                    ;PMMagDrv.asmの初期化ルーチンをコール
        goto        Mprompt                 ;モニタ・コマンド待ち
```

● ドライバ・プログラムの説明

CWおよびCCW信号は割り込みを実行させて1パルス分のモータ駆動を行います．割り込みを使わなくても目的を果たすプログラミングは十分に可能と思いますが，CWまたはCCW信号の発生からモータを駆動するまでの時間遅れをできるだけ均一にしたかったので割り込みを使いました．

図3-4を見てください．割り込みを使わない場合はこのようなフローになります．CWとCCW信号の検出を常にぐるぐると回していますから，信号が発生したときにプログラムはこのループのどこを実行していたかで反応する時間が左右します．この違いはせいぜい数マイクロ秒ですから，問題にする程度でもないのですが，理想に近づけようとして嫌ってみました．その理由は高速で回転中の安定性で，CWまたはCCWパルスが一定間隔で与えられても，その応答時間にばらつきがあると駆動は一定間隔でなく，結果的にギクシャクした動きになります．

図3-5が実施したプログラムのフローです．割り込み信号のサンプリングは4クロックで1ステートとした1ヵ所で行われるので，ワーストで4クロック分の遅れ時間の差はあるのですが，**図3-4**のようにループ・プログラムでポーリングするよりははるかに均一な遅れ時間になります．

リスト3-2が**図3-5**のフローで示した部分のプログラム・ソースです．CWまたはCCW駆動を行った後でCW/CCW入力が“H”になるまでループを5回繰り返しています．使用したPORT B割り込みは，状態の変化で割り込みが発生するので入力が“L”から“H”に戻ったときにも発生します．これを無視するためにCW/CCW入力が“H”になるまでの待ちを1回は必要ですが，5回にしているのはスイッチを手で操作した際のジッタによる誤動作を避けるためです．

図3-6は1ステップを動かすサブルーチンのフローです．2相励磁方式を採用したので4種類の励磁パターンをステップごとに変えるわけですが，変数StateCountの下位2ビットでパターンを選択します．CW方向ではStateCountをインクリメントして選択，CCWではデクリメントして選択です．

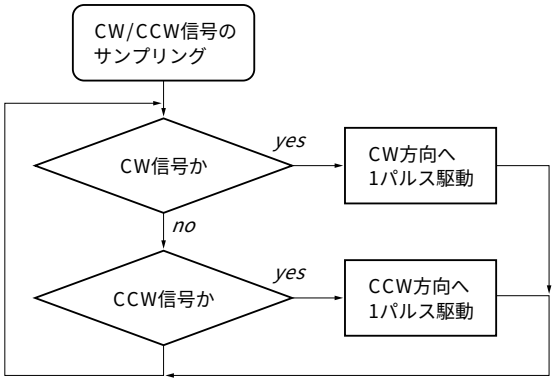

図3-4　あまりお勧めでないプログラム方法
ステッピング・モータの動作時間と比較すれば十分に高速で実行できるので実用的には支障ないはずだが，信号の発生から検出されるまでの時間が一定でないので嫌った．

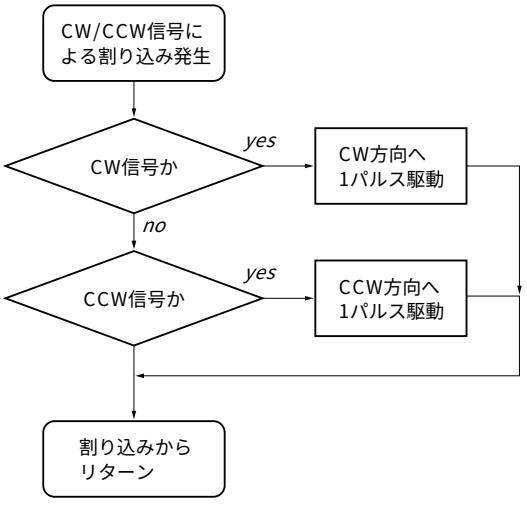

図3-5　実施したプログラムのフロー
今回プログラムした方法．CW/CCW信号で割り込みを発生させて，割り込み処理ルーチンで励磁パターンの切り替えを行う．

リスト3-2　割り込み処理のプログラム・ソース
ポートBのCW/CCW信号（RB_7,RB_6）の状態変化で割り込みが発生し，リスト先頭のラベルRBintで示したルーチンが起動される．

```
RBint  ;ポートB割り込みサービス
       movwf     wstac                ;wレジスタの待避する.
       swapf     STATUS,W             ;（zフラグが影響しない命令を使用）
       movwf     sstac                ;statusレジスタの待避する.
       btfss     INTCON,RBIF
       goto      RBie                 ;PORTB割り込みでないなら終了
       btfss     PORTB,7
       call      PMcw                 ; CW方向へ駆動
       btfss     PORTB,6
       call      PMccw                ; CCW方向へ駆動

       movlw     5                    ;cw/ccwが“H”になった確認の回数
       movf      waitcnt,F
RBie
       btfss     PORTB,7              ;cw/ccw入力が“H”になるまで待つ
       goto      RBie
       btfss     PORTB,6
       goto      RBie
       decf      waitcnt,F
       jnz       RBie                 ;確認の回数に達するまで繰り返す

       bcf       INTCON,RBIF          ;割り込みフラグをクリア
       swapf     sstac,W              ;（zフラグが影響しない命令を使用）
       movwf     STATUS               ;statusレジスタをもとへ戻す.
       swapf     wstac,F              ;（zフラグが影響しない命令を使用）
       swapf     wstac,W              ;wレジスタをもとに戻す
       retfie                         ;割り込みからリターン
```

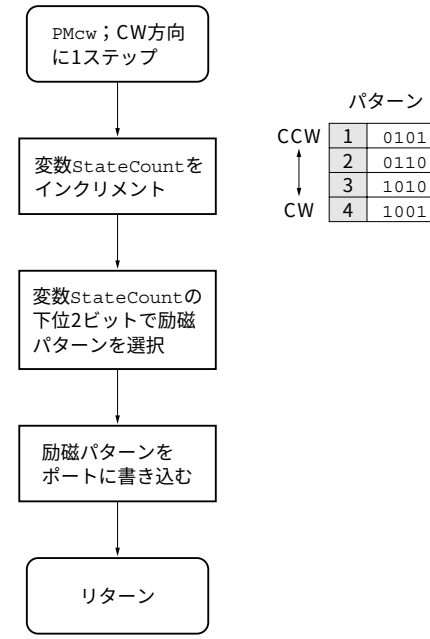

図3-6　CW方向に1ステップ駆動するフロー，サブルーチン名はPMcw
CCW方向ではStateCountをデクリメント，ほかは同一．

リスト3-3　励磁パターン読み出しのプログラム
パターンのテーブルがページをまたいだ場合でも支障ないようにプログラムされている．

```
MagPtn movwf     PCL
       retlw     B'00000101'          ;ステート1
       retlw     B'00000110'          ;ステート2
       retlw     B'00001010'          ;ステート3
       retlw     B'00001001'          ;ステート4


getptn ; StateCountの下位2ビット+1がoffset
       movf      StateCount,W
       andlw     3                    ;下位2ビットとand
       addlw     1                    ;+1
       movwf     offset               ;励磁パターン読み出しのオフセット
       movlw     LOW MagPtn
       addwf     offset,F             ;パターンのアドレスにオフセットを加算
       movlw     HIGH MagPtn
       btfsc     STATUS,C             ;ページをまたぐ?
       addlw     1                    ;またぐなら1+
       movwf     PCLATH
       movf      offset,W
       call      MagPtn
       return
```

励磁パターン読み出しのソースを**リスト3-3**に抜き出しました．getptnがそのサブルーチンです．四種類のパターンを引き出すルーチン“MagPtn”がプログラム領域のページをまたがなければ簡単ですが，またいだ場合にも備えてプログラムしてあります．パターンのアドレスにオフセットを加算した後で，キャリが発生したらアドレスの上位バイト“PCLATH”に1を加算します．

3-4　台形駆動で位置制御コントローラの製作

ステッピング・モータ・ドライバの用意ができたので，これを駆動して位置決めを行うコントローラを製作します．デバイスはここでもPIC16F84を使用しました．**写真3-2**が試作品です．位置制御の実行はコマンドとしてRS-232Cインターフェースで与えます．

このコントローラで駆動できるのは本章で紹介したドライバとは限りません．CW/CCW駆動の入力をもった市販されているドライバICやステッピング・モータ・ドライバあるいはサーボ・モータ・ドライバも同様に駆動できます．

● 回路

図3-7がコントローラの回路です．CN_1のCWはドライバ(**図3-3**)のCW，CN_1のCCWはドライバのCCWに接続します．(株)秋月電子通商のキットではCWを右回転のスイッチ，CCWは左回転のスイッチに接続します．

手持ちの都合でクロック発振には8MHzの水晶を使いましたが，セラミック振動子で十分です．周波数も8MHzにこだわる必要はありませんが，モータのスピードとRS-232Cのボー・レートに係わります．ボ