

実践報告・資料

ワンボードマイコンによる汎用旋盤の輪郭制御

青森職業訓練短期大学校 伊藤 祐規

An attempt to control the lathe by a microcomputer

Yuki Itou

要約 近年職業訓練カリキュラムにおいてメカトロニクスの分野が多く取り入れられている。メカトロニクスを学ぶ場合、機械的、電気的、ソフト的なものを同時に理解する必要上、身近にある具体的なものを対象とするとハードとソフトの関わりがよくわかる。従来マイコンやパソコンでステッピングモータを制御しグラフや絵を描く例は報告されている。ここでは、慣性が大きくバックラッシュを持っている汎用旋盤の切込みと送り軸にステッピングモータをとりつけ、輪郭制御による工作方式について研究し、実際に切削を行い、その実用性を検討した。例として球面削りを取り上げ、段削りを手動で行う場合と全く同じ手順で進め、送り量と切込み量は用意した実数計算のサブルーチンで求めている。また、輪郭制御は代数演算方式で用いている。

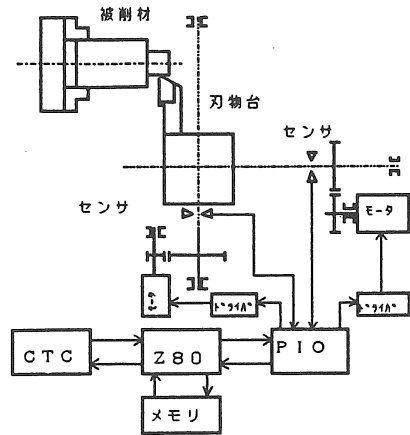
I はじめに

近年職業訓練カリキュラムにおいてメカトロニクスの分野が多く取り入れられている。メカトロニクスを学ぶ場合、機械的、電気的、ソフト的なものを同時に理解する必要上、身近にある具体的なものを対象とするとハードとソフトの関わりがよくわかる。そこでメカトロニクスを学ぶ実践課題として汎用旋盤の送り方向と切込み方向を制御する簡単なシステムを試作した。

このシステムでは、機械的なもの以外に次の知識が必要になる。

- 1) Z80 とそのペリフェラルのプログラミング
- 2) CP/M 上でのソフト開発
- 3) プログラムのモジュール化
- 4) プログラムのROM化
- 5) 浮動小数点演算
- 6) ステッピングモータとその駆動回路

図1に試作システムについての略図を示す。Z80 ワンボードマイコンからPIOを通じて2つのステッピングモータが接続され、1つは切込みハンドル軸、1つは往復台を移動させる送り竿に減速歯車を介して取り付けられている。刃物の位置決めはこの2つのモータを制御して行う。CTCはモータに送るパルス列の時間間隔(モータの回転速度)を設定し、2つのセンサは、各進行方向の原点を示す。切削条件の送り速度制御は、主軸



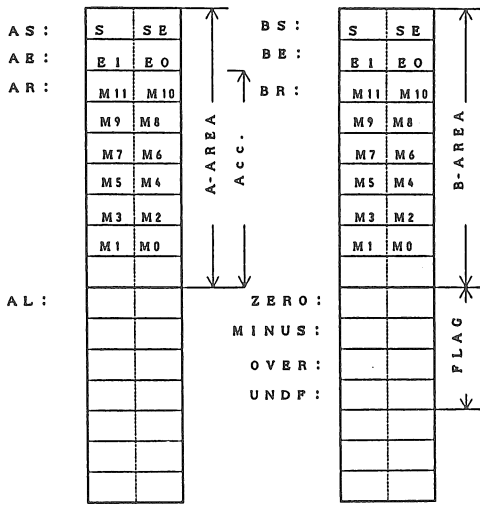
ワンボード	SHARP	SM-B-80TE
モータ	ORIENTAL	PH299-02B
		PH266L-01B
センサ	OMRON	EE-SX471

図1 システム略図

回転数がフィードバックされるようになっていない。設定した主軸回転数を一定とみて、それに対応した送り速度に固定される。軸方向の1mm移動は、切り込み側では100パルス、送り側では560パルスに対応する。したがって、実数で与えられた切込み量、送り量はそれぞれ100倍、560倍した後に整数化され移動に必要なパルス数に換算される。この計算は、浮動小数点演算で行う。

II. 浮動小数点演算

マイコンで行う実数の計算は、数の表現形式として2進数で取り扱う場合と、例えば、BCD 表現のように2進数でない場合に大別される。もちろん計算の速さ、計算に必要なメモリ数などから2進数表現形式の方が効率よく計算される。しかし、その数値計算プログラムを開発するデバック作業では、2進数表現は、それを見ても簡単にその値を理解することが困難である。ここでは、取扱が簡単なBCD表現を用い、CP/M上でのソフト開発手順を理解する目的で浮動小数点演算のサブルーチンを作成した。



S 仮数の符号 0:正 1:負
 SE 指数の符号 0:正 1:負
 E1, E0 指数の値 00-99
 M11, M0 仮数 M11キ0 M11とM10の間に
 小数点があるとみる。

図2 ワークエリア

図2に表現形式(フォーマット)とワークエリアの配置を示す。符号、指数で各1バイト、仮数部6バイト(12桁)である。このフォーマットでは、例えば S=0, SE=1, E1=0, E0=3, M11=1, M10=2であれば、 1.2×10^{-3} を示す。どの様な数でも零でない限り正規化され M11≠0となる。またワークエリアでは、A-AREAに被演算数、B-AREAに演算数が入り、演算結果はA-AREAに残ることとする。A-AREAの下部に8バイトの余裕を設けてあるのは乗算、除算で用いるためである。正規化の大まかな流れ図を次に示す。そして、次に示す加減乗除のサブルーチンを用意しておく。例えば、X番地からある数と、Y番地からある数とで

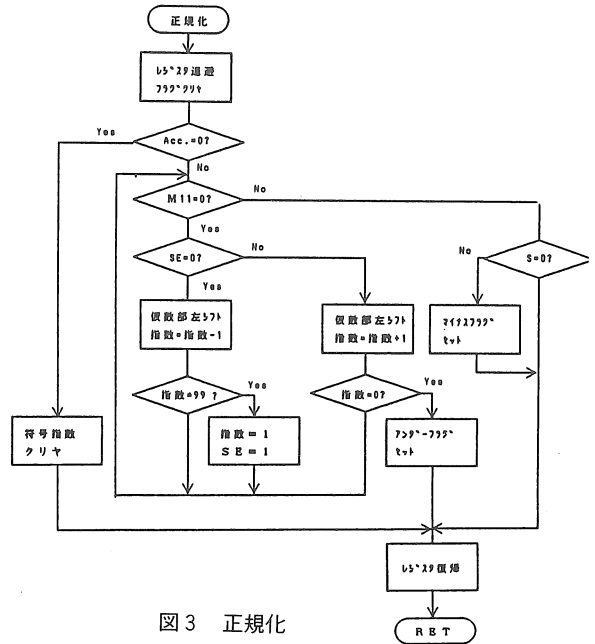


図3 正規化

	DE, HL レジスタに 数値の先頭アドレス を与えてコールする	HLに演算数の先頭 アドレスを与えて コールする
加算	A D	A D O
減算	S B	S B O
乗算	M U L	M U L O
除算	D I V	D I V O

$$Z = (X+Y) * Y$$

を求めたい時には

```
LD DE, X
LD HL, Y
CALL AD
CALL MULO
```

と実行すれば、A-AREAに計算結果が残る。上の8個のサブルーチンと、それを補助するサブルーチン群の大きさは、約1.4Kバイトである。

III パルス数と移動量

ここに使用した汎用旋盤の往復台の送り竿の回転に必要な起動トルクは、約6kgcmである。このため往復台の始動、停止にスローアップ、スローダウン制御を用いなければ十分な送り速度は得られない。この加速、減速に必要な移動量は、十分な距離を移動させる段削りにおいては、その内に含まれているため問題にならない。しか

し、輪郭制御の場合、制御対象となる1単位区間を位置決め可能な最小区間にするると加速、減速はできなくなる。そこで輪郭制御の場合、1単位区間を1/20mmとし、送り側で28パルスが対応するようにした。この28パルスの中で図4に示す階段状のスローアップ、スローダウンを行う。

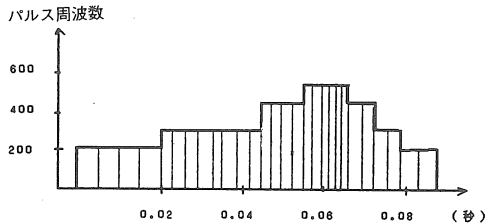


図4 スローアップ スローダウン

IV PIO と CTC の設定

図1に示すPIOとCTCの役割は、ドライバにパルスを送ることと、2つのセンサからの割り込み信号を受けることである。CTCの役割はドライバに送るパルス列のタイミングを割り込み信号でCPUに知らせることである。具体的には、CPUのシステムクロックが2.5MHzである場合

```
LD    A,0A7H    ;チャンネル制御語
OUT   (CTC0),A ;
LD    A,(TBL)   ;時定数
OUT   (CTC0),A ;
LD    A,00H     ;割り込みベクトル
OUT   (CTC0),A ;
```

とすると、TBL番地に格納されている値に応じて

$$T = \frac{256 \times (TBL)}{2.5 \times 10^6} \approx 0.1 \times (TBL) \text{ms}$$

T秒毎に割り込みを発生させるタイマモードとなる。

TBLの値を変えて上記の設定をやり直すことにより、スローアップ、スローダウンを実現する。次にPIOでは

```
LD    A,0FH     ;Aポート出力
OUT   (PIOAC)   ;
LD    A,0CFH    ;Bポートビットモード
OUT   (PIOBC),A ;
LD    A,03H     ;ビット指定
OUT   (PIOBC),A ;
LD    A,0B7H    ;割り込み制御語
OUT   (PIOBC),A ;
LD    A,0FCH    ;マスク指定
```

```
OUT   (PIOBC),A ;
LD    A,10H     ;割り込みベクトル
OUT   (PIOBC),A ;
```

とし、Aポートからの2つのドライバにパルスを出しBポートのビット0、ビット1にセンサからの割り込み信号が入るようにする。割り込みの優先順位は、PIOの方を高くする。

V バックラッシュの対応

このシステムの汎用旋盤の刃物台移動機構には、ボールネジ等を使用していないため反対方向の移動が起きる度にバックラッシュ処理の問題が生じる。前進後退の両方向に対して完全にバックラッシュを取り去ることは構造的に複雑になるのでここでは、一方方向に対してのみバックラッシュを取り去ることにする。図5に示す溝形のフォトマイクロセンサの溝内を刃物台と同じに移動する指針が通過する度に、1つのパルスが発生しこのパルスの立ち上がりで先に述べたPIOの設定により割り込みが発生する。往路と復路では少なくともフォトマイクロセンサの投受光窓以上のずれを生ずることとなる。しかし、段削りの場合であればプログラムにおいて、往路では、パルスカウントのリセット、復路ではスローダウンの開始になるようにしておくことによりバックラッシュの問題は解決されることになる。

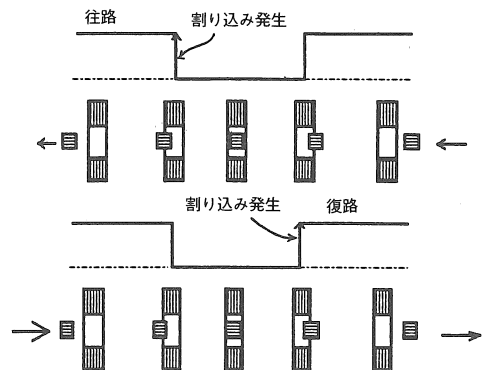


図5 センサ 往路と復路

VI 円弧補間 (代数演算方式)

図6に示すように、任意の格子点の座標を (x_i, y_j) とし判別式を

$$D_{i,j} = x_i^2 + y_j^2 - (x_0^2 + y_0^2)$$

とすれば、 $D_{i,j} > 0$ 、 $D_{i,j} = 0$ 、 $D_{i,j} < 0$ により点 (x_i, y_j) は、円弧の外側か円弧上か内側かにあることが判別できる。そこで始点 (x_0, y_0) から $D_{i,j} \geq 0$ であれば送り側で

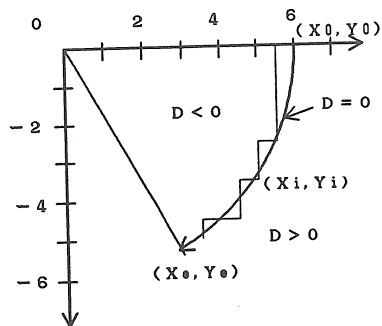


図6 円弧補間 (第4象限 右回り)

28パルス、 $D_{i,j} < 0$ であれば切込み側で5パルスそれぞれ発生させ移動する。そして x_i, y_j の値はそれぞれ

$$D_{i,j} \geq 0 \text{ ならば } x_i \rightarrow x_{i+1} = x_i - 1$$

$$D_{i,j} < 0 \text{ ならば } y_j \rightarrow y_{j+1} = y_j - 1$$

となり、新しい判別式はそれぞれ

$$D_{i+1,j} = D_{i,j} - 2x_i + 1$$

$$D_{i,j+1} = D_{i,j} - 2y_j + 1$$

となる。また終端条件は

$$x_i = x_e$$

$$y_j = y_e$$

である。

VII 例 (球面削り)

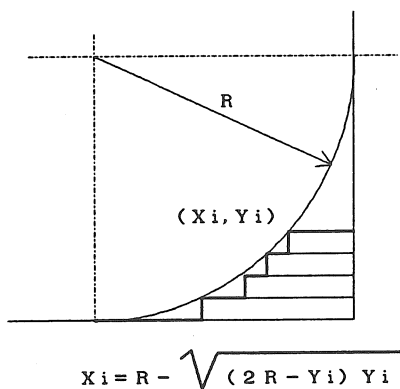


図7 球面削り

始めに端面を出しておき、そこを原点 (図7の右隅) と見なして指定された回数の段削りを繰り返す方式を採用した。すなわち、図7に示すように加工物の直径と段削りの回数を与えてやると、1回の切込み量と、切込み総量に応じて変化する送り量を計算し、段削りを繰り返す。そして球面削りに入る前に切り込み方向のバックラッシュを取り除き球面の中心を原点とみた座標値で第4象限右回りの円として円弧補間を応用した切削を行っ

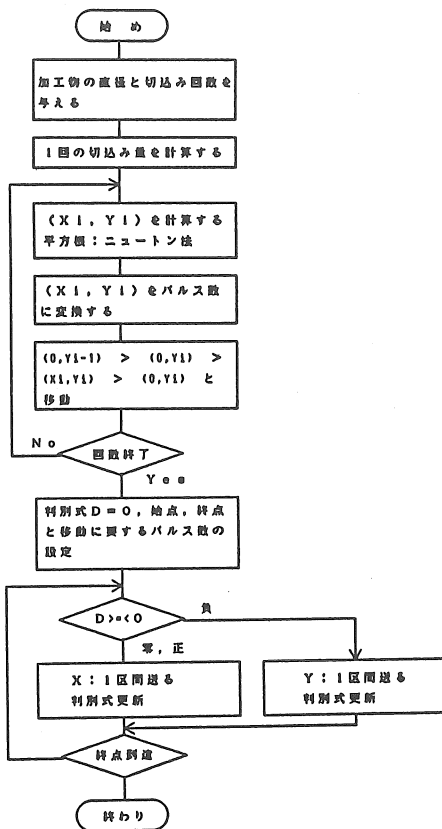


図8 球面削り

た。以上の流れを図8に示す。円弧補間でX方向に1区間送るとき28パルス移動してから判別式を更新するのではなく第1番目のパルスが出た直後に判別式の更新をして次の区間もX方向であれば、スローダウン、停止することなく定速のパルス列が持続するようにすると滑らかにモータを回転させることが可能であることが実証できた。

VIII おわりに

試作したシステムは、ハードとソフト面の結び付きと流れを理解するための実践的研究課題としては十分なものである。しかし使用したステッピングモータの能力から輪郭制御1区間は1/20mmであり現実的な加工精度を十分だし得ない等の問題 (球面に階段状に移動した工具軌跡がそのまま残る) は残っている。今後簡素で実用化に必要な手だてを考究しより効果的なシステムを目指し改良して行く必要がある。

参考文献

- (1) 横田 続 Z80 の使い方 オーム社 1983
- (2) 池辺 数値制御通論 オーム社 1985