

<開発・技術ノート>

## PLCプログラミング言語STによる制御理論(スライディングモード)の実装†

坂本 憲昭\*

### Realization of Sliding Mode Controller by using ST on PLC

NORIAKI SAKAMOTO\*

Although many researchers have proposed various control technology, it is difficult to apply these technologies to a real plant because of the shortage of software environment of the plant controller. In this report, the realization of self-tuning sliding mode controller on PLC (programmable logic controller) using ST (structured text) will be verified through experiments.

Key Words: IEC61131-3, ST, sliding mode control, self-tuning

#### 1. はじめに

多くの研究者により様々な制御理論が提案されているが、その理論を現実に稼働しているプロセス系に適用することは容易ではない。なぜなら、研究に使われるハードウェアはパソコン、ソフトウェアはC言語やVisual Basic(米国Microsoft社の登録商標)などが多いのに対し、現実に使われている制御装置は、PLC(Programmable Logic Controller)やDCS(Distributed Control System)がほとんどであり、ラダー図言語がメーカー固有のプログラミング言語で制御則を記述しなければならない。このような理由から、開発した制御理論を移植することが容易で、かつ現場に受け入れられる制御機器が望まれる。

この要望に対し、製造現場という環境を考慮してPLCを選択すれば、パソコンから移植する際の開発効率を考慮してC言語<sup>9)</sup>またはBASIC言語<sup>10)</sup>をプログラミング言語として使用可能なPLCを考える。これらはメーカーが単独に提供しており機種別の制約を受けるが適用事例も公表されている<sup>11)</sup>。

一方最近では機種(メーカー)を限定しない国際規格のプログラミング言語のひとつとしてST(Structured Text: 構造化テキスト言語)が公開された。数社のメーカー(松下電工、富士電機、三菱電機、横河電機、日立、シーメンス、ロックウェル、シュネデールエレクトリック。2003年8月現在。筆者の個人的調査にて不十分な点があることをお許し願う)がサポートしており、制御理論をこのSTでプログラミングして実現することが有効と考えている。しかしながら、STを用いた制御事例の紹介は海外ではあるが<sup>2), 4)</sup>、国内では皆無である。

そこで本稿の目的は、具体的な制御理論として筆者らが研究しているセルフチューニング型のスライディングモード制御を取り上げてプロセス系の簡単な実験を行い、この有効性を評価することにある。

#### 2. STについて

ST<sup>3), 5)</sup>は、PLCのソフトウェアオープン化に対応するため国際標準プログラミング言語IEC61131-3(JIS B 3503)<sup>12)</sup>で規格化された5つの言語、

- ・LD(Ladder Diagram: ラダー図言語)
- ・FBD(Function Block Diagram: ファンクションブロックダイアグラム言語)
- ・IL(Instruction List: 命令リスト言語)
- ・SFC(Sequential Function Chart: シーケンシャルファンクションチャート)
- ・ST(Structured Text: 構造化テキスト言語)

のひとつである。この規格化の目的は、

- ・機種に依存しない
- ・構造化プログラミングにより保守しやすい
- ・再利用性の高い部品化
- ・厳密な文法チェックによるバグの減少

などにある。STはテキスト型のプログラミング言語であり、IF文、CASE文、繰り返しなどの命令をもつ。三角関数、指数、対数、平方根、絶対値などの算術関数をサポートし、3次元までの配列、構造体、関数の定義と呼び出しが提供されている。さらに記述はC言語に似ており、C言語で開発した制御理論を移植しやすい。参考のためC言語とSTのおもな命令比較を付録に示す。

#### 3. セルフチューニング型スライディングモード

##### 3.1 セルフチューニング型スライディングモード制御

スライディングモード制御(Sliding Mode Control, 以後SMC)は、パラメータ変動や外乱などに対して強いロバスト性をもつ。これまで数多くの実際のメカニカルなシステムや実プロセスにも適用され、その効果が確認されている。

しかしながらプロセス系への適用事例は少ない。理由のひとつはSMCが制御対象のモデルを要することである。そこで、セルフチューニング型のSMCが検討されている<sup>6)~8)</sup>。

† SICE Annual Conference in Fukui 2003で発表

\* 職業能力開発総合大学校電子工学科  
(〒229-1196 神奈川県相模原市橋本台4-1-1)  
Polytechnic University  
(Received August 7, 2003)

この制御則は対象を限定するが、調整係数が少なく制御対象のパラメータを必要としない。紙面の都合で操作量  $u$  の算出式のみを示す。

$$u = -YA_E + \gamma c \varepsilon \cdot \text{sgn}(c\varepsilon) \quad (1)$$

$$Y = [(1-c)\dot{\varepsilon} - \alpha x], \quad \dot{A}_E = [(c-1)\dot{\varepsilon} + \alpha c\varepsilon, -c\varepsilon x]$$

ここで、 $c$  調整係数、 $\varepsilon$  偏差、 $\alpha$  目標値の微分値、 $x$  制御量である。 $\gamma$  の値は制御中に指数平滑法で求める<sup>8)</sup>。

### 3.2 スライディングモード制御の課題

SMC を適用する場合、前述のセルフチューニング型に限らず操作量を高速に切り換える必要があるため、制御装置の早い応答が要求される。切り換え遅れは操作量の激しいチャタリングを誘発する。

高速な切り換えを実現するために、アナログ回路や専用 LSI (ASIC)<sup>1)</sup> による制御則の実現が提案されている。これらの専用回路には劣るが、PLCの特徴に高速性(高速命令、高速処理、高速応答、高速スキャン)があり、PLCならば早い切り換えが期待できる。さらに、専用回路に比べて制御則のゲイン調整やプログラムの変更が容易という付加価値がある。

## 4. 実験

Photo.1 に示す PLC (松下電工(株))、左側から電源、CPU + アナログ I/O) を用いて、セルフチューニング型 SMC の実験を行う。



Photo.1

### 4.1 実験概要

1 水槽の水位制御 (Fig.1) とラバーヒーターの表面温度制御 (Fig.2) を行う。目標値は共に

$$x = at + \beta \quad (x: \text{水位または表面温度}, t: \text{時間}) \quad (2)$$

としてパターンで与え、外乱として水位制御では下部の排水バルブを開閉し、表面温度制御ではランダムに冷風の吹き付けを行う。

### 4.2 STによるプログラミング

プログラムの一部を Fig.3 に示す。文末のセミコロンや変数の型形式が厳密な点など C 言語に近い。ここで、目標値は時間関数であり ST でも計算が可能だが、目標値の計算を他の研究ですでに IL にて作成しており、今回これに手を加

えることなく再利用した。このようにサブルーチンのような処理のまとめ (これを POU と称する) 毎にプログラミング言語 (2章に示した5つの言語) を選択できるので、IL に限らず PLC で多用されているラダーの既存プログラムをそのまま再利用することもできる。また変数の通用範囲についても C 言語と同様に各 POU が参照できるグローバル変数と、POU 内でのみ有効なローカル変数に分かれる。開発方法の詳細は例えば文献 3) を参照していただきたい。

参考までに本実験の開発環境の画面を Fig.4 に示す。Fig.4 の左枠 POU の下 JIKOKU, SELF\_VSC, SV\_JIKOKU はサブルーチンであり (すなわち POU が3つ)、プログラム言語は上から IL, ST, IL を選択している。右枠上段には、各サブルーチンが読み書き可能なグローバル変数を、中・下段には IL と ST のソースコードを表示した。

### 4.3 C言語からの移植性

開発効率の程度を示すため一例を述べる。本実験の制御則はパソコンで C 言語にてシミュレーションを行っている。そ

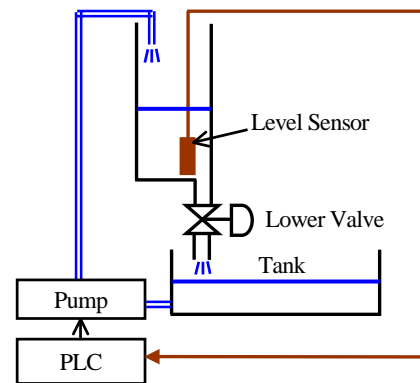


Fig.1 Experimental water level control

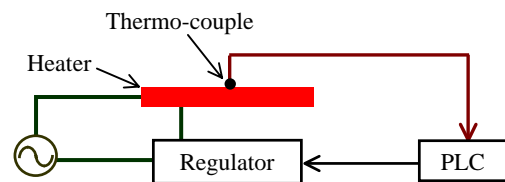


Fig.2 Experimental heater control

```

rSUM1:=rSUM1+rSAT*rSigma*rErr;
rSUM2:=rSUM2+rSAT*rSigma;

rU:=rUeq+(rSUM1+rAnufa*rSUM2)*rK*rSgn;

(* MAX-MIN CHECK *)
IF rU < -10.0 THEN
    rU:=-10.0;
END_IF;

```

Fig.3 ST Programming

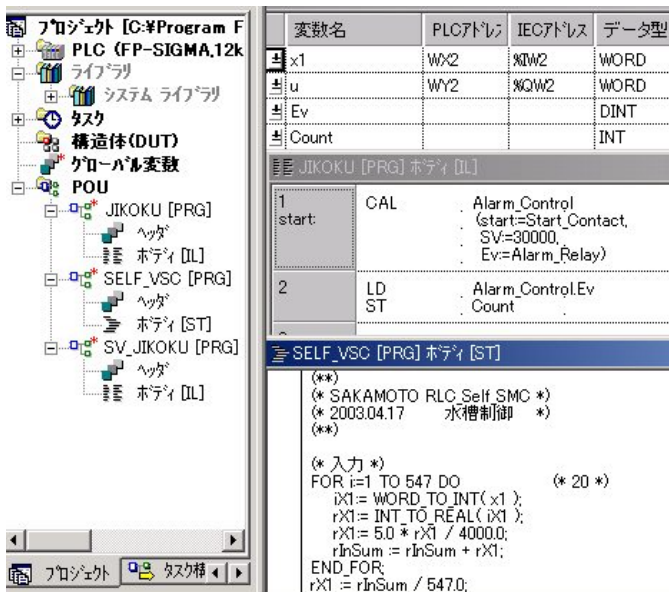


Fig.4 Development environment

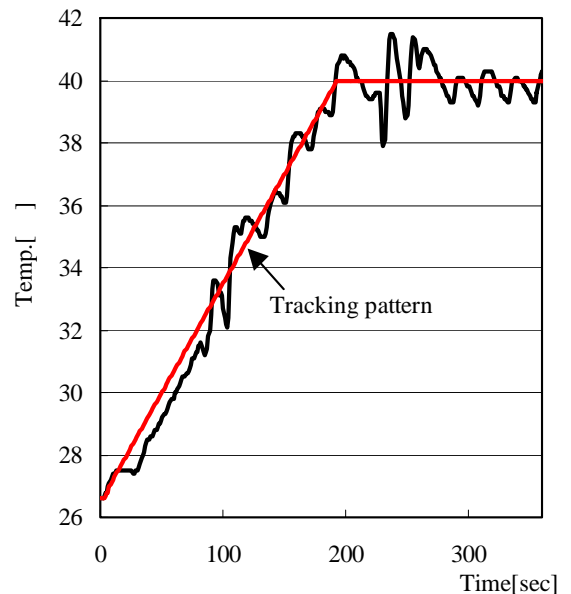


Fig.6 Experiment Results (Heater Control)

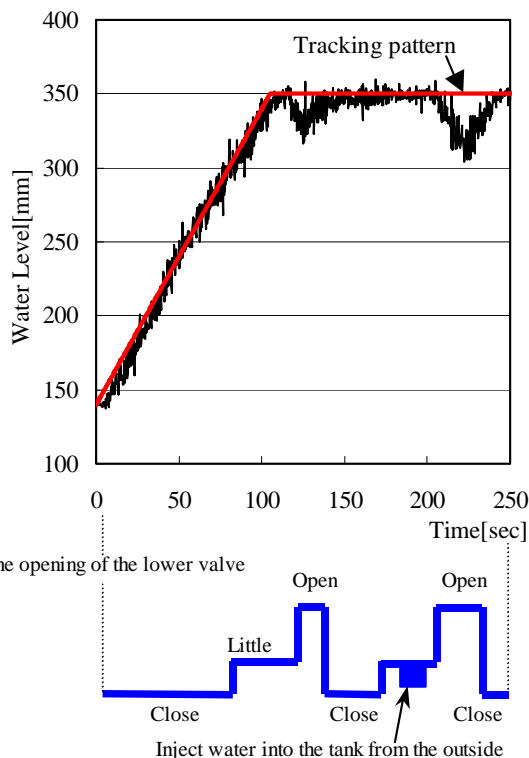


Fig.5 Experiment Results (Water Level Control)

の制御則のC言語サブルーチン（制御量と目標値から操作量を算出する）は 36 行である．このサブルーチンをSTに変換するのに約 5 分，次にFig.4 に示した開発環境の使用は初めてにもかかわらず，入出力変数の定義，他サブルーチンの組み込みとデバックを 1 日で完了した．このようにC言語で研究された制御理論を，ラダーを使用せずに簡単にPLCに実装可

能である．またFig.4 のような開発環境は各メーカーとも提供している<sup>12)</sup>．

#### 4.4 実験結果

水位制御の実験結果を Fig.5 に示す．図中の上段が制御結果であり，下段は外乱として与えた下部にある排水バルブの開閉履歴である．温度制御の実験結果を Fig.6 に示す．温度の振れは外乱によるものである．両図とも制御量が外乱による影響を受けているが，目標パターンに拘束されており，制御則が有効に働いていることがわかる．

#### 4.5 制御周期

水位制御のプログラムステップ数は 1136，制御周期は 4.23 msec，そのばらつきは標準偏差で 0.37 msec であった（測定結果が妥当な数値であることをメーカーに確認している．その回答を付録 に示す（メーカー了解済み））．本実験ではこのばらつきが SMC に対して影響を与えなかったが，例えば応答が速いプロセスを制御する場合を考えれば，セルフチューニング型の SMC が要求する制御周期について今後検討する必要がある．

### 5. おわりに

ST によるプログラミングは，C 言語からの移植が容易であり，制御理論を実装しやすいことを確認した．その一例として，高速な応答を要求する SMC を実現した．さらに，パソコンにはない PLC の下記の特徴が現場に受け入れられやすい．

- ・制御中に変数のモニタやプログラムの変更が可能
- ・既設のラダー図言語プログラムの活用が可能
- ・既設機器との接続性に優れる
- ・装置が小型化されている

・耐環境性に優れる

短所として、入力や出力などの各種データを制御中にモニタできるが、ファイルとして保存できない。事後解析するためには、データレコーダに別途出力するか、ネットワークやWeb対応機能の追加によるデータ転送が必要となる。

もちろん高速アナログ処理 (0.5 msec ~ 1 msec 周期) を可能にする PLC もございますが、コスト的には同機種の数倍 ~ 十数倍になります。コストと性能のバランスの問題ですので、この点をご容赦願います。

(松下電工(株) 制御分社 NAISFAN コールセンター)

付録 (C と ST のおもな命令比較)

|        | C   | ST   |
|--------|---|--|
| 注釈文    | // -----<br>/* -----*/  | (* ----- *)  |
| データ型   | int   | INT  |
|        | long int  | DINT   |
|        | float   | REAL   |
|        | char  | STRING   |
|        | bool  | BOOL   |
| 配列     | Array[5]  | Array[5]   |
| 代入文    | =   | :=   |
| IF 文   | if ( a>=0 ) {<br>処理;<br>}<br>else{<br>処理;<br>}                                | IF a>=0 THEN<br>処理;<br>ELSE<br>処理;<br>END_IF;          |
| FOR 文  | for( i=0; i<=100; i++) {<br>処理;<br>}  | FOR i:=0 TO 100 DO<br>処理;<br>END_FOR;                  |
| CASE 文 | switch (a) {<br>case 0: 処理; break;<br>case 1: 処理; break;<br>default: 処理;<br>} | CASE a OF<br>0: 処理;<br>1: 処理;<br>ELSE 処理;<br>END_CASE; |

参考文献

- 1) 福島, 他: FA ロボット特集 モーションコントロール ASIC,R&D / 神戸製鋼技法, 43-1, 69(1993)
- 2) M.Kunz and G.Suss: SPS-Sprachnorm programmiert Antriebssteuerung, Elektronik, 1, 80/83 (1994) (in German)
- 3) 関口隆: 新しいプログラマブルコントローラのプログラミング IEC61131-3 による効率的プログラミング, ISBN4-339-03171-2, コロナ社 (1999)
- 4) J.Bartels and B.Scherff: Automatisierungskonzepte im Vergleich, Elektronik, 11, 50/60 (1999) (in German)
- 5) NAIS Control FPCWIN Pro 導入ガイド, 松下電工(株) (2001)
- 6) 坂本憲昭: セルフチューニング型スライディングモードコントローラ, 第 44 回自動制御連合講演会前刷, 714, 584/585 (2001)
- 7) 坂本憲昭: ST(IEC61131-3)による制御理論の実装について, 第 31 回制御理論シンポジウム, 葉山 湘南国際村, 331/334 (2002)
- 8) N.Sakamoto: Estimate of Disturbances for Self-Tuning Sliding Mode Controller - Realization of the controller by using ST (IEC61131-3) -, SICE Annual Conference in Fukui 2003, OS-8, 福井大学 (2003)
- 9) オムロン(株): SYSMAC ACE,  
[www.plcsoft.ne.jp/soft/Jpn/Info/ACE/ACEPiDefault\\_J.shtml](http://www.plcsoft.ne.jp/soft/Jpn/Info/ACE/ACEPiDefault_J.shtml)
- 10) 横河電機(株): FA-M3, BASIC CPU モジュール,  
[www.yokogawa.co.jp/IA/fam3/products/bp20.htm](http://www.yokogawa.co.jp/IA/fam3/products/bp20.htm)
- 11) 江口, 藤中, 野口, 小崎, 浪江: PLC を用いたバッチ反応機のモデル予測制御, 計装, 45-7, 21/25 (2002)
- 12) PLCopen (欧州に本部を持つ IEC61131-3 の普及団体)  
[www.plcopen-japan.jp/tech.html](http://www.plcopen-japan.jp/tech.html)

付録 (メーカーからの回答)

実験に使用された PLC (FP シグマおよびアナログ I/O ユニット) の性能上、計測された結果とほぼ同じになるかと存じます。

もともと同機種は小型 PLC ですので、高速アナログ処理を目的とする製品ではありません。おもに温度や圧力等のプロセス制御やインバータ制御を安価に実現できることを狙いに作られています。(筆者補足: 電源, CPU, アナログ I/O ユニットで定価 ¥ 118,900)