

# PLC 計装における再利用性と品質向上

東京電機産業

菊池 勇記

## 1. はじめに

PLC と汎用 SCADA で構成される PLC 計装は、以前に比べ格段にエンジニアリング効率と信頼性が上がってきた。PLC はラダー言語から Tag No ベースの IEC 言語へ、SCADA は Export/Import 機能が充実し、グラフィックへの割付も Drag & Drop で行えるようになった。またユーザのニーズに伴い、PLC 計装として冗長化を備える機種も登場し、従来のプロセス制御システムの市場にも適用されだしている。

プロセス制御の分野に適用されるに従い、ユーザからは DCS 並みのシステム信頼性を求められるようになってきた。

そこで我々は、CPU 及び制御 LAN の二重化に対応したコントローラであり、且つオンラインメンテナンスが可能な横河電機製 自律型コントローラ FCN/FCJ と SCADA としてはコントローラとの一貫性を考慮し STARDOM ASTMAC を採用することにより、上記のようなユーザニーズに対応している。

本稿では PLC 計装(STARDOM 計装)における、アプリケーションソフトの再利用性と品質向上を目的としたエンジニアリング手法を紹介する。



## 2. PLC 計装の問題点

PLC 計装はオープン性が非常に高く、様々なメーカーの SCADA と PLC が存在し、その組み合わせは多様である。その反面マルチベンダを採用すると SCADA と PLC が一体となっていない為、PLC 側と SCADA 側の両方でデバイス(変数)のエンジニアリングが必要となり、二度手間となることがある。

例えば、PLC 側でデバイスの変更があった場合、SCADA 側でもデバイス割付の変更が発生する為、PLC-SCADA 間のインターフェイスのエンジニアリングが非常に重要となっている。

このような問題を解決する為に、我々は PLC 側のデバイス等に依存することなく SCADA 側を構築することができれば SCADA ソフトの再利用性が高まると同時にエンジニアリングの品質も高まることが可能だと考えた。そこで、SCADA 及び PLC で構造体の概念を持たせることで、デバイス等の固有情報に依存せず、SCADA 側の構築が行える手法を考案した。

### 3 構造体手法への取り組み

構造体とは一般的に様々な種類のデータから互いに関連するデータをまとめた集合体を示すが、我々は PLC-SCADA 間のデータの取り合いに構造体データを使用することを検討した。

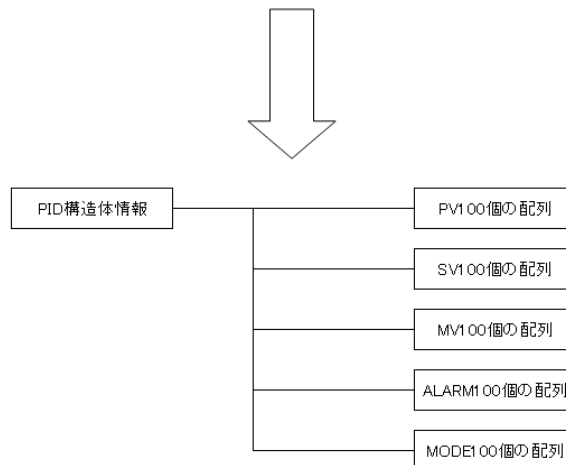
我々が採用している SCADA である横河電機製の STARDOM ASTMAC にはブロックデータオブジェクト(以下 BD)と呼ばれる構造体の概念をもったデータタグが存在する。この BD では容易に効率よく大量データの送受信が行える機能を有し、さらに構造体の一部の要素のみの送受信もサポートされている。

また、同じく横河電機製の自律型コントローラである FCN/FCJ は、国際標準規格「IEC61131-3」に準拠しており、従来の PLC で扱うデバイスとは異なり、コントローラ内部に構造体の配列変数を作成することができる。

SCADA である ASTMAC の BD はこのコントローラ側の構造体配列変数にアクセスすることができる為、コントローラ側と SCADA 側の構造体データをそれぞれ関連付けすることで、コントローラのデバイス等の固有情報に依存せず SCADA 側の構築が行えることとなる。

構造体とは、「お互いに関連するいろいろな種類のデータをまとめて、1つのかたまりにしたもの」である。  
PID計器が100個あり、画面に表示する情報としてPV値、SV値、MV値、ALARM状態、MODEが必要な場合、

PID01	PID02	PID03	PID04	PID05	PID06	.....	PID99	PID100
PV	PV	PV	PV	PV	PV	.....	PV	PV
SV	SV	SV	SV	SV	SV	.....	SV	SV
MV	MV	MV	MV	MV	MV	.....	MV	MV
ALARM	ALARM	ALARM	ALARM	ALARM	ALARM	.....	ALARM	ALARM
MODE	MODE	MODE	MODE	MODE	MODE	.....	MODE	MODE



以下に STARDOM 計装を用いた場合についてももう少し詳細な内容を説明する。

#### (1) 従来手法

コントローラ上に定義した制御ブロックのデータを ASTMAC のグラフィック画面に貼り付ける場合、自動的に作成される csv ファイルをインポートし、グラフィック画面に指定する必要があった。

この場合、コントローラ側での定義が変更となった場合には再度インポートを行い、さらにグラフィック画面上の割付も変更することとなる為、手間がかかりまた間違いを起しやすかった。

## (2) 構造体手法

まずコントローラ側で構造体を定義し、必要な制御ブロックデータを構造体変数へ格納する。ASTMAC 側では BD を定義し、BD が参照するコントローラ側の構造体変数を指定する。グラフィック画面では BD の信号名称を指定するため、コントローラ側の変更があっても、ASTMAC のグラフィック画面での修正は不要となる。

さらにこの方式の場合、グラフィック画面毎に BD データをまとめて読み込める定義が可能であるため、一括で画面上のデータを読み込むことができ、通信パフォーマンスの向上という観点からも非常に有効な手法である。

また、この手法の場合、同一設備向けのレポート案件などではグラフィック画面の再利用が可能となり、システム構築の品質向上に貢献している。

### ASTMAC構造体定義

```
PV(100) ,single
SV(100) ,single
MV(100) ,single
MODE(100),integer
ALRM(100),integer
```

合計1600Byte

```
ASTMAC オブジェクト名定義
名称   BD01
型     BD
アドレス @GV.udtStruct01
```

### FCN構造体定義

```
TYPE
REAL100 : ARRAY[1..100] OF REAL;
REAL50  : ARRAY[1..50]  OF REAL;
INT100  : ARRAY[1..100] OF INT;
INT50   : ARRAY[1..50]  OF INT;
END_TYPE
```

```
TYPE
stData:
STRUCT
PV      : REAL100;
SV      : REAL100;
MV      : REAL100;
MODE    : INT100;
ALRM    : INT100;
END_STRUCT;
END_TYPE
```

合計1600Byte

### FCNグローバル変数名定義

```
変数名  udtStruct01
型       stData
```

## 4. STARDOM コントローラの特徴

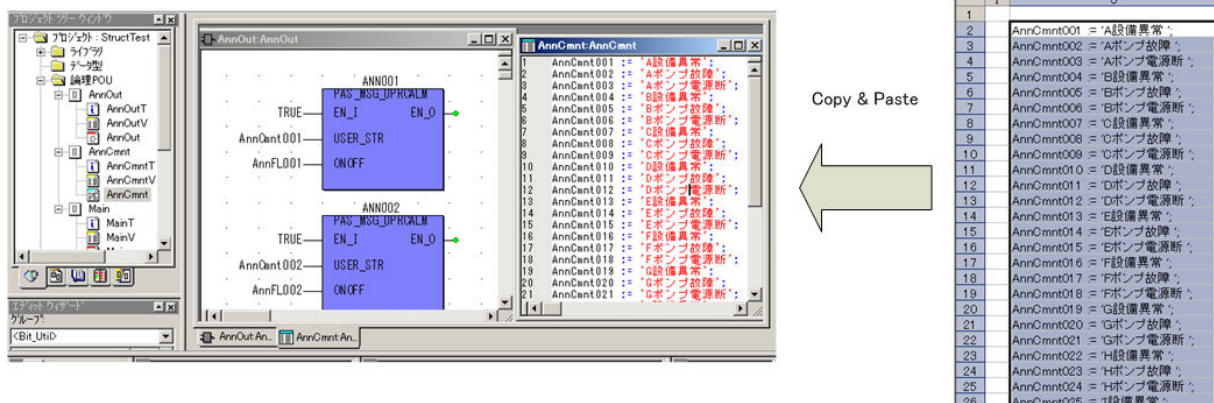
STARDOM 計装は構造体の扱いに長けていることを説明したが、そのほかの特徴についても少し説明する。

自律型コントローラ FCN/FCJ は「オンラインダウンロード」がサポートされており、機能的な連続性が保たれたままのプログラム変更が行える。また PC 上で動作する「シミュレータ」が用意されており、開発及びデバッグ時には実機がなくてもプログラムの動作確認が行える。更に横河電機製の DCS とも接続することが可能となっている。DCS の HMI に SIOS(System Integration OPC Station)を介してFCN/FCJと接続することにより、DCS 上の HMI で FCN/FCJ の制御ブロックを操作・監視することが可能となっている。

## 5. アナシエータ機能の充実

PLC 計装では DCS と異なり、メッセージの発報を SCADA 側にて管理される事が多い。異常信号等を PLC へ取り込み、その異常信号と接続するオブジェクトを SCADA 側で定義する。更にメッセージ出力を行う設定とメッセージコメントを設定し初めてメッセージ出力用オブジェクトとなり、アナシエータの点数と同数のメッセージ出力用オブジェクトを作成する必要がある。

しかし STARDOM 計装ではメッセージ定義はコントローラで行うことが可能となっている。メッセージ出力は保持型と非保持型が用意されているのでアラームとオペガイドの様な使い分けが容易に行えるのである。また、メッセージコメントを変数にすることにより定義も非常に容易に行える。



ツールの説明となってしまうが、FBD(ファンクションブロックダイアグラム)以外に ST(ストラクチャードテキスト)を使用すると、マイクロソフト社の Microsoft Excel 等でコピーしたものをそのまま貼り付けることが可能である。これを活用することにより各種コメント部分のエンジニアリング効率は大幅に向上する。

## 6. 品質の安定に向けて

これまで、SCADA 側のエンジニアリング効率改造方法について記述してきたが PLC 本来の目的であるシーケンス制御についても工夫をし、品質の安定を図った。

一般的にステップシーケンスが遷移するエンジニアリングを行う際には、工程と呼ばれる区分を作成しその工程を組み合わせてシーケンスを完成させて行く。我々はその工程の考え方の見直しを行った。

制御要素 (バルブやポンプ) から工程を考える事により、設備に属する制御要素への複数工程からの競合をなくし、増改造への対応も容易に行えるのではないかと考えた。

複数工程から制御要素へアクセスを許可してしまうと、同時に複数工程が実行中の場合にどの工程によって制御要素が操作されたのかを判断するのが困難となり何か不適合が発生した場合に問題の解決が遅れる。

新しい工程区分の考え方のポイントとして、

- ・ 制御要素は 1 つの設備区分のみに属し、複数設備区分で操作しない。
- ・ 1 つの工程区分の中で複数の実行シーケンスが働かないようにする。

ことが挙げられる。

仮に他設備区分の制御要素に対する動作指令を行う場合は直接操作するのではなく、動作指令要求フラグを 1Shot で出力されるように設計した。そのフラグを受信した設備区分でその制御要素が動作可能かを判断し、実際に操作を行う。

このように他とのインターフェースも明確にした工程区分を設計することで、改造が容易になるだけでなく、予期しない動作を未然に防ぐ事ができ、品質の安定も図ることができる。

## 7. まとめ

IEC でのプログラミングは容易に機能の部品化(ファンクションブロック化)が行え、構造体や配列を用いたエンジニアリング手法は再利用性を大幅に増加させる。我々はそれに合わせたASTMACの定型化を行うことにより、STARDOM計装の標準化を行う事に成功した。今後もその標準機能をブラッシュアップし、更なる機能改善及び品質向上を目指して行く所存である。

キクチ・ユウキ

東京電機産業株式会社 システム ENG 部

〒151-0072 東京都渋谷区幡ヶ谷 1-18-12

電話(03)3481-1130

E-mail : [yuuki-kikuchi@tokyo-densan.co.jp](mailto:yuuki-kikuchi@tokyo-densan.co.jp)