

箱焼鈍炉計装システムへのオブジェクト指向設計の適用

Application of Object-oriented Design Method to BAF Instrumentation Control System

関口 修⁽¹⁾ 大 力 修⁽²⁾ 中 島 雄 二⁽³⁾ 田 原 伸 一 郎⁽⁴⁾
Osamu SEKIGUCHI Osamu DAIRIKI Yuji NAKASHIMA Shin'ichiro TAHARA

抄 録

近年、計算機ソフトウェアにおけるオブジェクト指向言語に基づいた設計技術は、従来の手続き型言語によるものと比較して、保守性、再利用性の面で格段に優れていることから、ビジネスユースや、製造計画システム等の構築時に積極的に適用されつつある。これらオブジェクト指向の利点をプラント制御システムに反映すべく、オブジェクト指向技術による制御システム構築ツール"GOOD"を開発し、新日本製鐵八幡製鐵所のBAF計装制御システムに適用した。本システムは、ホットラン後約1年を経過したが、順調に稼働を続けている。

Abstract

In recent years, design method based on an object-oriented language in the computer software is actively being applied in constructing business systems and scheduling systems such as host computers and manufacturing planning system and so on, since it is superior to the traditional one based on a procedural language in respect of maintainability and reusability. In order to put the benefits of this method to the production control system engineering in a plant, we developed a construction tool for the control system based on the object-oriented method, called "GOOD", and applied it to construct a real instrumentation control system for box annealing furnace in Yawata Works, Nippon steel Corp., called "BAF". About a year has run since its hot-run, and the system is achieving good performance and reliability.

1. 緒 言

今から約20年前に登場した計装DDC(Direct Digital Control)は、それ以前のハード回路による計装制御機能を計算機によるソフトウェア処理に置き換えた、極めて機能的自由度の高い制御システムである。

しかしながら、この優れた自由度の反面、従来の計装制御では調節計などのハードの組み合わせによって機能設計していたものが、“データ”と“処理”という形で分離されたために、本来の計装設計にはあまり本質的ではない、データアドレス、メモリ割付などに気を使いながら、システムを構築することが必要になった。この傾向は、連続制御とシーケンス制御が混在する系において更に顕著となり、計装エンジニアリングにおける設計の考え方を大きく変える必要が生じた。

その後、計装制御システムの主流は、ミニコンピュータからマイクロコンピュータへと進化、DDCからDCS(Distributed Control System)へと発展し、ソフトウェアの設計用ツールも充実していったが、やはり“データ”と“処理”が分離している以上、本質的解決策とはならなかった。

我々は、新世代のDCSを設計するにあたり、この点を最も重視した。つまり、計算機のデータ処理形態に依存しない、本来の計装制御系設計に合った形のDCSソフトウェア構築環境の実現を目指した。そのためには、オブジェクト指向設計は不可欠のものであった。

2. オブジェクト指向設計適用の考え方

プロセスの動的制御にかかわる連続制御と操業における運転手順を実現するシーケンス制御に分けて考える。

連続制御の設計においては、従来の計装フローシートイメージが重要なため、この単位でのオブジェクトを機能構成の基本要素とする。つまり従来のハードウェアコンポーネンツイメージ(調節計等)をオブジェクトとしてソフトウェアコンポーネンツ化し、このオブジェクト間のつながりを入出力データメッセージで結合することで連続制御のループを構成する。制御モードの変更や、制御開始・終了タイミング、設定値の自動変更等は、各オブジェクトへの変更要請メッセージの形で指令し、オブジェクト自身がこれを実行することで、各コンポーネンツの独立性を確保し、機能変更を容易にしている。

⁽¹⁾ 八幡製鐵所 設備部 部長代理

⁽²⁾ エレクトロニクス・情報通信事業部

システム研究開発センター 所長

⁽³⁾ ニッテツ八幡エンジニアリング(株)

システムインテグレーション・グループ 部長

⁽⁴⁾ ニッテツ八幡エンジニアリング(株)

電計エンジニアリング・グループ チーフエンジニア

また、オブジェクト指向の特徴的な機能としての“継承”と“集約”を利用し、例えば、制御機能のループ単位での継承や機能単位での集約を行うことで、燃焼制御オブジェクトや、炉圧制御オブジェクトといった大きな単位での再利用を可能とし、システム構築における大幅な生産性向上を実現した(図1参照)。

次にシーケンス制御では、従来の処理手順を記述したシーケンス制御イメージのプログラムをすべて無くすることが必要となる。この理由は、シーケンス機能をロジック部分のみのプログラムとして記述した場合、運転における手順と、関連する連続制御の機能やデータが分離された形となり、制御機能全体としての把握が困難となるためである。我々は、シーケンスは本来運転方案を実現するものであり、これを忠実に実装できること、すなわち、起動条件(イベント)と、運転条件(インターロック等)により、対象の動作を規定できることが必要であると考えた。このために、本システムでは、オブジェクト指向設計の大きな機能であるメッセージリンクと、このリンク上にメッセージを走らせるための条件テーブルの方式を採用した。この機能により起動タイミングの発生元と発生条件、各種運転条件の記述を可能にし、同時に、これらを連続制御系設計のフローシート上に実装することで、起動すべき制御オブジェクトに直接作用させることが可能となり、連続制御、シーケンス制御を含め、制御機能のすべてを計装フローシートのみで構築できるようになった(図2参照)。

また、従来の計装エンジニアによるオブジェクト指向設計及びソフトウェア製作を可能とするためには、GUI(Graphical User Interface)

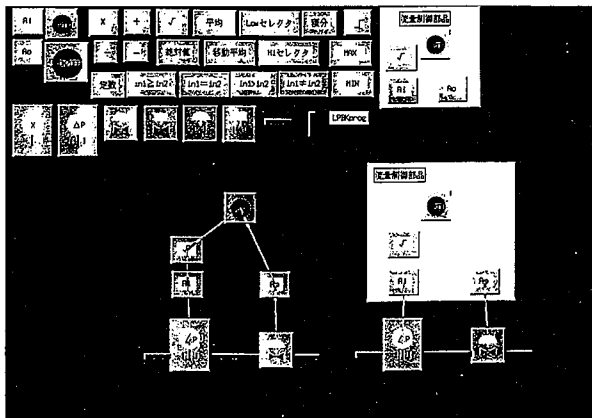


図1 オブジェクト指向設計による連続制御系設計例

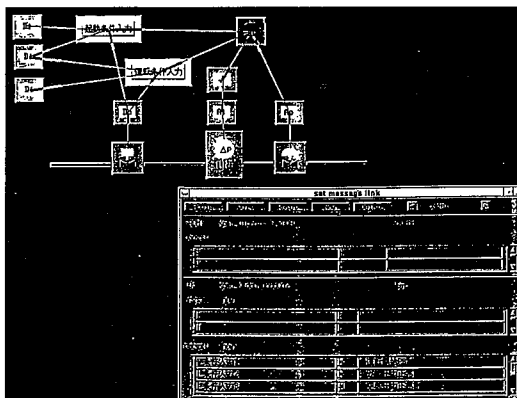


図2 オブジェクト指向設計による起動条件、運転条件の設定例

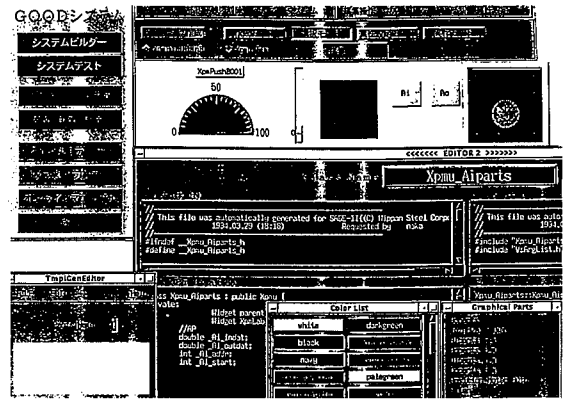


図3 GOOD(Global Object Oriented Designer)システム

によるビジュアルプログラミングの実現が不可欠であったが、当時のエレクトロニクス研究所(現エレクトロニクス・情報通信事業部システム研究開発センター)で、オブジェクト指向設計・製作環境として開発中であったシステムの豊富なGUI機能を活用、加えて、プロセス制御向けに機能を強化することにより、現行のGOOD(Global Object Oriented Designer)システムを完成させた(図3参照)。

3. GOODシステム

本章では、開発したGOODシステムの仕組みや機能について説明する。

3.1 動作環境

具体的動作環境としては、安価であることとハード依存性を極力無くすることを目的として、UNIX[®]系オープンシステムを採用した。

・ハードウェア

CPU部：ワークステーション(WS)または、パーソナルコンピュータ(PC)

Pio部：汎用シーケンサ

・OS

Solaris 2.x(WS)

SunOS 4.1.x(WS)

SVR4(WS)

Linux(PC 1996年12月より)

PANIX[®]2(PC 1996年12月より)

・GUI

Motif[®]3(WS, PC)

・コンパイラ

C++ ver3.1以上

3.2 制御システム作成の仕組み

システムの作成はすべてウィンドウ上で行う。まずメニューからシステムビルダを選択すると、前述の計装制御用の各種オブジェクトを内蔵したツールが起動される。ここで今回の制御システムを作成するために白紙のウィンドウを開く。

次に、必要なオブジェクトを計装フローシートを作成する要領で配置して行く。すなわち、計装機能設計において最も本質的な要素であるフローシートの作成を進めて行くことにより、自動的にそのオブジェクトがプログラム上に配置されて行くのである(図4参照)。

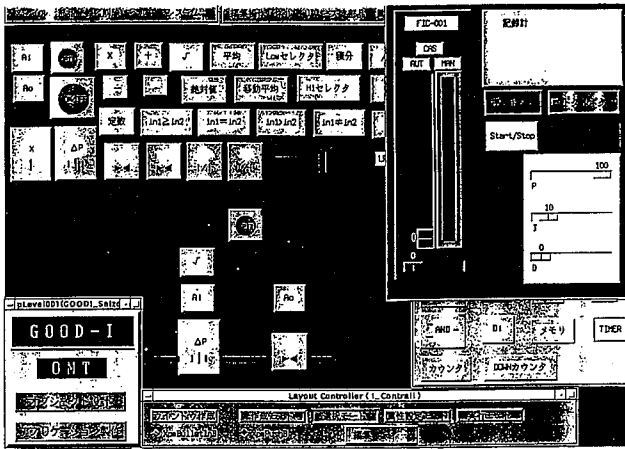


図4 計装制御オブジェクトの実装

しかし、このままでは、各オブジェクトがただ存在するだけで動作しない。そこでメッセージリンク機構を用いて、各オブジェクト間の結合(=プログラム起動情報を定義)をメッセージで与え、オブジェクトを起動させる。通常のフィードバック制御であれば、各コンポーネントの出力を入力側から順に出力側に向かってリンク接続し、各処理結果のみを次のオブジェクトへ出力すればよい。制御の開始メッセージは、タイマーにより一定周期で、各オブジェクトへ指示され、連続制御は定周期で実行されることになる(図5参照)。

このメッセージリンクによりメッセージ発信の条件を記述したり、メッセージの内容をその条件で変更することができる。従ってバッチ制御や、処理内容によるモード、パラメータの自動変更など、運転方案に関連する機能は、すべてこのメッセージリンク機構で実現できることになる。すなわち、このシステムにおいては、記述が起動条件や運転条件といった機能を直接的に指し示すものであり、従来のプログラムにはなかった点、更に、計装フローシート上で記述できるので、データアドレスやプログラム命令を探す必要がない点が特徴となる。

図6に、燃焼制御における点火から昇温消火までの機能を、制御オブジェクトとメッセージリンク機構で構築している例を示す。例えば、図左上の加熱開始ボタンが押されたときに、これに関連する動作を起こすのは、それぞれリンク線で接続されたPIDコントロー

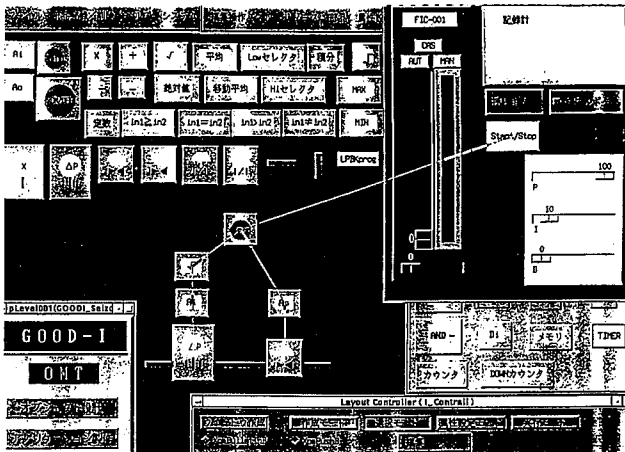


図5 メッセージリンクによるオブジェクト間の結合

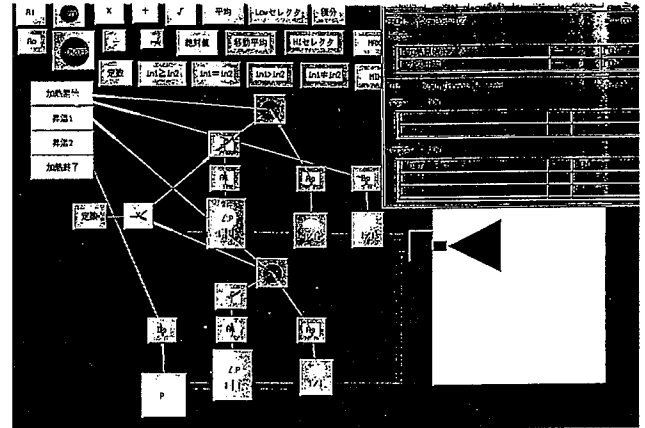


図6 燃焼制御系の設計、実装例

ラと遮断弁、燃焼プロアであり、図右上のテーブルは、このときに、ガス圧力の状況を見て、燃焼プロアに起動をかける条件を記述している。このようにすべてがこのフローシート上表現され、かつ運転方案部分のイベントと条件が分かりやすい形で実装されている。

3.3 システムのテスト

システムのテストは通常、プログラム動作レベルのテストと、ハードを含めた総合試験とで行う。

まずプログラム動作レベルのテストでは、制御対象をシミュレートするオブジェクト(一次遅れ、無駄時間等)を計装フローシート上に配置し、メッセージリンクで制御装置オブジェクトと接続する。この状態で、実際の運転条件に極めて近いテストを連続して行うことにより、制御にかかわる機能のバグ対応は、この段階でほとんど完了し、総合試験を効率良く推進することができる(図7参照)。

また、更新等で、旧システムから短期間で新システムへ移行しなければならないような場合、このシミュレーション機能が威力を発揮することになる。

3.4 機能変更・追加

GOODシステムでは、作成したシステムをその場で直ちに動作させることができるため、制御機能の作成途中でも制御機能を実際に確認しながら設計を進めることができる。機能変更や追加は、できるだけこのフェーズで吸収する。しかしながら、現実のプラント計装システムでは、現地調整段階で種々の変更や追加が発生することがあり、この場合、変更・追加による影響が他の部分に影響しない

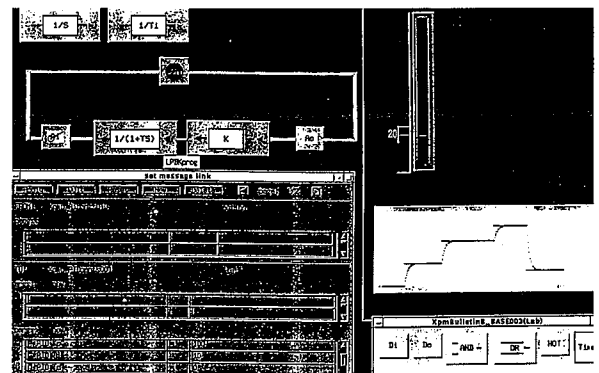


図7 制御状況のシミュレーション

ことが望ましい。しかし、これまでのシステムでは、その構造上、プログラムレベルでの影響が避けられなかった。

これに対してGOODでは、オブジェクト指向設計されているため、オブジェクトの追加とメッセージリンクの変更により機能変更を行うことができるので、他のオブジェクトに影響を与えることは少ない。

4. BAFへの適用とその評価

新日本製鐵八幡製鐵所BAF(Box Annealing Furnace)雰囲気制御システム更新にGOODシステムを初めて適用した。本件を例として実機適用における技術的ポイントと評価及び今後の展開について述べる。

4.1 システム構成

BAF雰囲気制御システムに必要なとされる主機能は、

- ・ガス流量制御
 - ・ガス自動切替
 - ・圧力・温度監視
 - ・上位計算機(プロセスコンピュータ)からの設定値受信による全自動運転
 - ・上位計算機(プロセスコンピュータ)への実績送信
- であり、ガス流量制御、ガス自動切替、圧力・温度監視についてはスタンド各々に対して行う。また、これに対するハードウェアの構成は以下の通りである。
- ・計装制御用ワークステーション：SPARC 75MHz, MEM 64MB
 - ・RPio：汎用シーケンサ
 - ・計装用CRT：i486DX2 DOS/V, MEM 32MB

ハードウェア構成を図8に、操作監視用画面を図9に示す。

4.2 適用における技術的ポイント

制御のループ数が非常に多く、各流量制御ガス切替機能が付属する形になっているので、これを基本部品の組み合わせで全部作成する

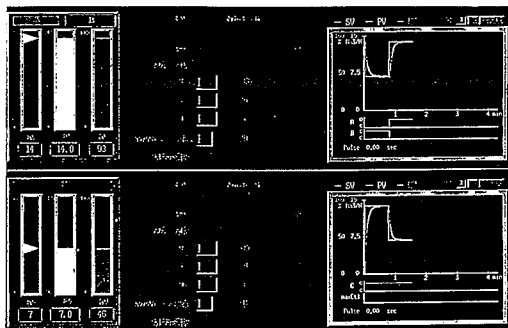
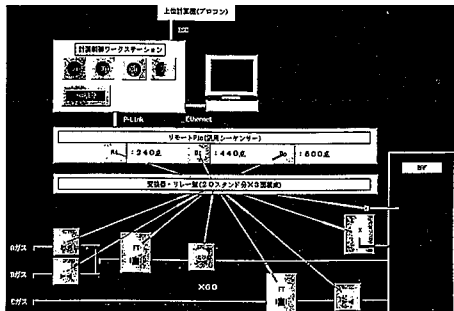


図9 BAF雰囲気制御システム操作監視画面

と、かなりの製作負荷となる上にメンテナンスも困難となる。そこで、下記の手順で基本部品を統合した新しいオブジェクトを作った(図10参照)。

基本部品→集約→流量制御部品

→継承後ガス切替機能追加→新流量制御部品

新流量制御部品→継承後各ループごとの属性追加、各スタンド分インスタンス作成

もう一つのポイントはメモリの有効利用である。GUIを用いて作成した制御機能は、メンテナンス性に優れる反面、常にグラフィックスを伴っているため、非常に大きなメモリ量を消費する。そこで、制御の実行に不要なGUIを、実行モジュール作成のコンパイル時に削除する機能を自動生成機能に付加し、GUI無しコンパイルを可能とした。これによりメモリ消費量は5分の1まで削減できた。

4.3 評価

制御性能については、対象としたプロセスに対する計装制御システムとしては、下記のように満足できる応答性と制御精度を確保すると同時に、高いソフトウェア生産性を実現することに成功した。

- ・制御周期：約120ループを1.5秒周期で制御
 - ・制御精度：1%以内(整定後設定値に対して)
 - ・ソフトウェア生産性：従来比1.3倍
- 従来型DCSとGOODでのシステム化の詳細比較を表1に示す。

5. 今後の展開

現在は、ハードウェアのダウンサイジング化を更に推進するため、パーソナルコンピュータでの開発を急ピッチで進めている。対象とするパーソナルコンピュータは、耐環境性を考慮し、DOS/V系

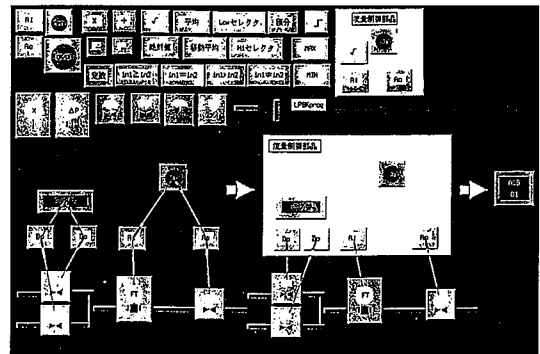


図10 BAF雰囲気制御における制御オブジェクトの継承と集約

表1 従来型DCSとGOODシステムの比較

評価項目	従来型DCS	GOOD	
ソフトウェア	生産性	1.0	1.3
	流用性	異なるメーカー間での流用困難	異なるメーカー間でも流用可能
ドキュメント	流用性	同上	同上
教育期間		1か月程度	1～2週間程度

のFA用機種を選択している。現時点(1997年2月)でGOOD本体の移植は完了し、制御用各種オブジェクトを移植中である。今後、例えば8ループ単位での分散制御システム化開発を行い、GOODシステムも本格的な分散型のDCSへと展開して行く計画である。

また、パーソナルコンピュータ、ワークステーションともにCPU性能が格段に進歩しており、最近(1997年1月)のシミュレーション結果において、6連のテーブル速度設定制御と、矯正機のロール位置設定制御を50ms程度の制御周期で実行できることを確認、簡単な電気制御系への適用も可能な状況となっている。

また、機能的には、汎用データベースの結合、並びに、物流制

御用の各種オブジェクトを開発中であり、システムインテグレーションのプラットフォーム的存在として発展させて行く方針である。

*¹UNIXは、X/Openカンパニーリミテッドがライセンスしている米国並びに他の国における登録商標。

*²PANIXは、エー・アイ・ソフト株式会社の商品名。

*³Motifは、Open Software Foundationの登録商標。

その他会社名、製品名などは、製造会社及び販売会社の商標若しくは登録商標である。