

Masayoshi SUGIMOTO and Yuuki KAWARASAKI

Department of Physics, Japan Atomic Energy Research Institute

## ABSTRACT

The JAERI free electron laser oscillation program requires a small and flexible system for controlling and monitoring of various kinds of parameters. Three major components, injector, accelerator and optical device, in the facility are independently controlled by the local control units, which are operated by personal computers and CAMAC crate systems. These are integrated, by dual network, to construct a distributed control system. The modularity of hardware and software of the control system is the key issue of the design. The multiple user interfaces are provided to accommodate the development and the operation by users with different background.

## 原研FEL制御システムの設計

## 1. はじめに

原研の自由電子レーザー (FEL) 発振計画では構成機器として (1) 電子銃、SHB、バンチャー、前段加速部からなる入射部 (2) 超電導加速空洞、クライオスタット系、RF系からなる加速部 および (3) アンジュレータを含むレーザー光学部の3つの主要部分がある。個々の部分自体の規模はそれほど大きくはないものの制御すべき対象・内容はそれぞれ特徴的であり多岐にわたる。また、今後のR&Dの進展いかんでは現在想定している機器の細部について大きな変更を迫られる可能性も大きい。したがってFEL用の制御・計測系は多種のパラメータを取り扱うことができるうえに機器の変更やシステムの組み替えに対して容易に対応ができるものでなければならない。

制御系にたいする要求としては上記のフレキシビリティに加え、少々のパラメータの変動についてはそれに追従して発振が維持できるくらいのスピード (リアルタイムネス) も重要である。また実験機であるため一般に非常に多くのパラメータをレーザー発振の間 (1msec 幅、10pps) だけ一時にモニターすることになる。つまりシステム負荷が時間的に偏るので全体の構成は複数のCPUによる分散制御とし、かつトランスペアレントなネットワーク構成にしておき負荷の分散化をはかる。またネットワークは2重にしておきそのトラフィックも負荷分散が可能なようにしておく。

以下では具体的な設計に先立ち基本概念を検討した結果として、FEL制御・計測系に要求される性能を満たすために必要とされるハードウェアとソフトウェアについて述べる。特にシステムの開発・運転に多くのユーザーが容易に関われるように複数のユーザーインターフェイスを用意しておく必要があり、それらについて幾つかの提案を述べる。

## 2. 基本概念

## 2.1 ハードウェア

上述したように全系は大きく3ブロックに分けて考えることができ、各ブロックは原則的にそれぞれのローカル制御ユニットで制御・計測がなされる。ひとつのローカルユニットはパーソナルコンピュータをホストにしたCAMACクレートシステムを中心に構成される。FEL装置では、電子ビームとレーザー光の双方を複合的・有機的に制御しモニターしなければならず加速器制御及び放射線計測に使用経験が長く各種モジュールの揃っているCAMACシステムがいまのところ適任であると判断される。

個々のローカルユニットは主として自分の持ち場の制御と計測を司ることになるが全

体系を統一的に操作するためにネットワーク構成をとる必要がある。FEL装置の規模は小さなものなので通常のLANにこだわることなくいろいろな選択が考えられる。さらに上述のようにシステム負荷が集中することを避けるためにネットワークは2重化しておく。以下に幾つかの項目についてさらに考察する。

#### ○ ホストコンピュータ

コスト性能比の面からローカルユニットのホストコンピュータはパーソナルコンピュータを採用する。CPUはインテル 80x86 とする。その理由は並行プロセスの実行を素早くおこなうこと（プロテクトモード）と過去に開発したソフトウェア資源（リアルモード）を活用するためである。

運転制御用コンソールは独立にもう一台のローカルユニットを設けのホストコンピュータをコンソール専用にするかもしくは3台あるローカルユニットのどれかが代行できるようにする。コンソール用に一台確保する場合はローカルデバイスとしてコントロール・パネル、データベース・サブシステム等が考えられる。

#### ○ CAMACシステム

各ローカルユニットの制御対象範囲はかなり限定されているので原則として一台のCAMACクレートがスタンドアロンで用いられそのクレートコントローラはホストコンピュータとのバス直結型となる。ローカルユニット自体もできるだけ負荷を分散するためGPIB(IEEE-488)等の標準インターフェースバスを用いて末端機器の計測・制御のインテリジェント化をはかる。

#### ○ ネットワーク

FEL装置の規模は小さいので通常使用されるLAN(Ethernet や Token-ring)の形態に必ずしも固執する必要はない。運転用コンソールとして独立のコンピュータを採用するならばスター型トポロジーで構成することもありうる。いずれにせよネットワークの一時的なトラフィックスの負荷の増大をやわらげるためまたバックアップの意味もかねて2重構成とする。これらのネットワーク媒体、トポロジーは必ずしも同一のものである必要はなく、プロトコル層の低レベルでは個々のネットワークは独立に動作する。両者をどう使い分けるかはソフトウェアの問題になる。

### 2. 2ソフトウェア

FEL装置の制御・計測系をうまく動作させるための最大のキーポイントはソフトウェアである。ハードウェアの性能向上は日進月歩であり装置の完成時には今の10倍程度の改善が期待できる。したがってソフトウェア側では将来のハードウェア変更に対応できるものを準備しなければならない。そのためには徹底したモジュール化と階層化を追求することになる。

一般にはコンピュータのオペレーティングシステム(OS)はそのシステムの最大公約数的な使い方に合致するべくつくられているが逆にいうとシステムの細部にわたる操作がむずかしい上に冗長な部分も多くなる。その意味でFEL装置用のOSはその機能要素がプロセスの形で実現される(プロセス指向)構成にしユーザープロセスとの垣根をとり自在にシステムの変更ができるようにしておくことが望ましい。このような思想を小規模コンピュータシステムにとりいれたものにForthがある。Forth自身は一種のプログラム言語であるが文法上ほかの言語とかなり異なった要素を多く含むためひろく利用されているとはいえないものの小さなシステムでの応用としては非常に魅力あるものといえる。現実には開発用に複数の言語(アセンブラ, C, Forth, Pascal, Fortran, Lisp, Prolog, Smalltalk 等)を目的に応じて使用できるようなサポートを考えておく必要がある。

#### ○ ネットワークプロトコル

2重ネットワークのプロトコルは上層レベルでは同一のものでありユーザ側からはひとつに見えるが各に物理層の異なるネットワーク形態を用いた場合にはそれぞれの使い分けが問題となる。例をあげると(1)一方をバックアップシステムとみなす、(2)メッセージの長さ(情報量)により混んでいない(負荷を増やさない)方を使う、(3)一方を単純なメッセージ専用にする。FEL装置の場合は試験機なのでバックアップを第一義に考える必要はなくメッセージもある程度決まり切った単純なものが主であるためダイナミックにパスを振り分けるよりは(3)の形が最も効率的と考えられる。

具体的な案としてはひとつをEthernetタイプのバス・トポロジーに、もう一方をスター・トポロジーにとり、前者はファイル転送を後者は単純メッセージ転送に用いる。単純メッセージの中身は制御コマンドとステータス回答でありおおむね1パケットですべての通信が終了するようなものである。ときにはこれをパブリックメッセージとし、ローカルユニットどうしの同期シグナルとして利用することもあろう。かたやファイル転送としてはローカルシステムのログ・レポート、データベース・アクセス、システム・リコンフィギュレーション等が考えられる。ただしシステム負荷が増大したときにはオペレータの定める優先度にしたがって単純メッセージによるアクセス権割り当てがなされ、1対1通信的な転送が行われるようにする。なお2つのネットワークの使用はOSにまかせることもユーザプロセスが指定することも可能である。また転送プロトコルもユーザ選択が可能な何種類かを用意する。

### 2.3 ユーザーインターフェース

FEL装置の場合、ユーザーは運転者でありかつ開発者であるわけで両方の役割がうまくブレンドされた形のユーザーインターフェースが望まれる。システム構成はかなり頻繁に変更が加えられる可能性大であるため、いわゆる(ハードとしての)コントロール・パネルに類するものは最小限度に押さえる。そのかわり、大部分の表示・操作はグラフィクス・ディスプレイ上のアイコンにより行う。つまりコンソール機能の殆どはソフトウェアでシミュレートされる。ローカルユニットのユーザーインターフェースも同一であるのでネットワークのどこからでも共通の制御をすることが可能である。

ユーザーインターフェースとしては上記の(1)グラフィクスオリエンテッドなものほかに(2)キーボードからのコマンド入力(コマンドシェル)による方式、(3)表形式入力(スプレッドシート)による方式、(4)データベース(過去の運転データと理論・設計計算プログラム)を用いたガイダンス入力(エキスパートシステム)方式が用意され必要に応じてユーザーが選択し組み合わせて利用する。

### 3. おわりに

FEL装置は電子ビームの入射・加速・輸送といった従来の加速器制御の面と光学系にかかわる計測・解析の面とが複合的にからみあっている。このため制御システムとしては個々の異質な部分毎の制御をローカルに分離して行う一方それらを有機的に結合して、最終的に的確に現象をとらえられるよう情報を整理して提供することが強く要求される。結論として制御システムは分散処理・集中監視方式をとり、ハードウェア・ソフトウェア両面のモジュール化をはかったものとする必要がある。これにより今後の継続的なグレードアップにも柔軟に対処できる。またユーザーインターフェースはシステムの開発と運転が統一的に行いやすいものであるべきで、グラフィクスオリエンテッドなものを中心に複数のタイプのものが準備されその中からユーザーが状況に応じて選択が可能にする。これによってユーザー自身が自分で特殊な運転モードの開発や次の改良のステップへの糸口を容易に見出せることが期待される。