

PRESENT STATUS AND UPGRADE OF KEK-PF CONTROL SYSTEM

Takashi Obina¹, Ceol-on Pak, Yasunori Tanimoto, Takashi Nogami, Kentaro Harada
Photon Factory, Institute of Materials Structure Science
High Energy Accelerator Research Organization (KEK)
1-1 Oho, Tsukuba, Ibaraki 305-0801

Abstract

In the KEK Photon Factory (KEK-PF), upgrade project of its straight sections is underway. As a part of the project, control system is partially upgraded to the EPICS based control system. In this paper, present status of the PF control system and the detail of the renewal procedure is reported.

KEK-PFリング制御システムの現状と更新

1. はじめに

現在KEK-PFリングでは直線部増強のための改造が進行中であり、2005年9月末からの運転開始へ向けて作業を行っている^[1]。制御システムに関しては、既に1997年の高輝度化の際にミニコンピュータベースの旧システムからワークステーションを基にした制御系への置き換えが行われているが、今回の改造では更に制御の利便性と各種制御プログラム開発を容易かつ短期間に実現するために、EPICS (Experimental Physics and Industrial Control System)^[2]の一部導入を進めている。本稿では現在の制御システムの概要を説明するとともに、EPICS導入の状況、アーカイブとデータ取り出し、制御CPUとして使用するVME-CPUボードの割り込み応答等について述べる。

2. PF制御システムの現状

2.1 ハードウェア

現在の制御システム概略を図1に示す（詳細については参考文献[3]を参照）。基幹ネットワークには主スイッチと12台のスイッチングハブからなるATM/Ethernet(155Mbps)を使用している。現状でこの帯域がボトルネックになっているということは無いが、より大量のデータ転送に対応するために今年度中に主スイッチをGiga-switchに更新する予定である。制御サーバー計算機にはヒューレット・パッカード社製のワークステーション(HP-UX 11.11)を使用し、RF、マグネット、入射の制御タスクや、後述するデータチャンネルサーバー等はこのサーバー計算機上で動作している。研究所のネットワークとの間にはFirewallが導入されている。

2.2 デバイス制御

使用しているフィールドバスは真空、RF、マグネットなどのグループによって異なる。真空制御系

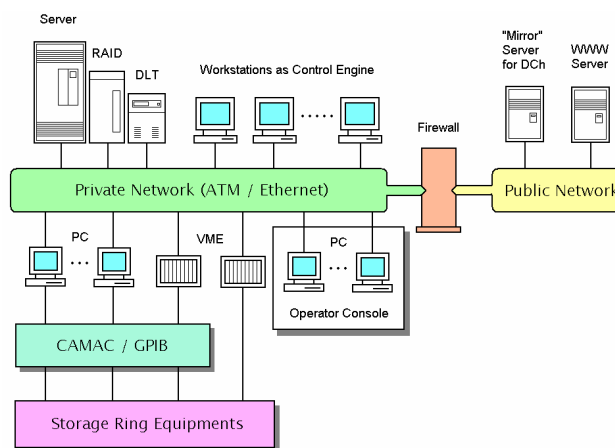


図1：PF制御システム概略(現状)。ATMスイッチはギガビットスイッチへ更新する。

の更新に関しては参考文献[1]に詳しく記述してあるので、ここではRF制御を例に説明する。RF制御のフィールドバスには主にCAMACが使用されている。使用しているボード(Input Register, Output Register, Control Registerなど)はPF建設当時からのもので、6台のクレートはパラレルハイウェイで接続されている。高輝度化以前は富士通製ミニコンピュータ(S3500)からの直接制御であったが、1997年にPCをデバイスサーバとして採用し、ワークステーション上で動く制御タスク(FORTRANによって記述されている)と、PCとの間の通信はネットワーク経由でのTCP/IPソケット通信で行う方針として汎用のサーバープログラムの開発を行った(図2)。その結果、デバイスサーバはPCのみではなくVME-SBC(single board computer)でも動作可能となっていて、リアルタイム性の要求精度に応じて使い分けられている。当然ながら、デバイスサーバを入れ替えても上位の制御プログラムは変更不要である。また、複数クライアントの同時接続も可能となっている。

¹ E-mail: takashi.obina@kek.jp

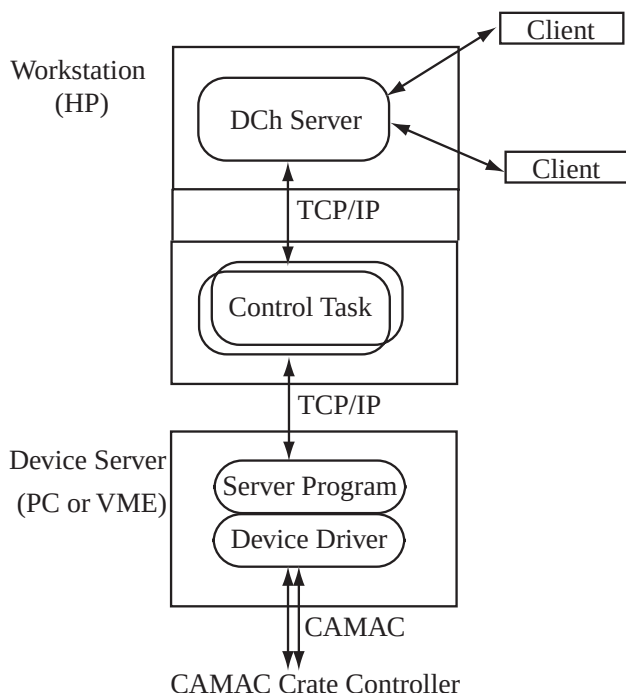


図2：通信経路/データチャンネル(DCh)概略。

2.3 データチャンネル

各種のデータを様々なプログラムから共有するためにデータチャンネル(DCh)と呼ばれるシステムの開発を行った^[4]。これはワークステーション上に作成した共有メモリと、TCP/IP経由でクライアントから読み書きするためのライブラリから成る。通常のBSDソケットプログラムと同様にクライアントはサーバーのIPアドレス/ポートを指定して接続し、その後はメンバー名とチャンネル名を指定することで各種データにアクセス可能となる。例えばビーム電流を読む場合はgeneral:current、RF周波数ならばrf:freqといったキーワードを指定してデータを読み出す。単純なプロトコルながら高速であり、Unix/Windowsなど各種のプラットフォーム上で動作するため、簡単なGUIはVisual BasicやDelphi等のRADツールで作成して運転に使用している。また、共有メモリのミラーサーバーを設置することにより、研究所内のネットワークヘデータを供給している(図1右側のMirror Server)。

通常の運転時は常時120 - 150個程度のクライアントが接続してデータのやりとりを行っているが、ワークステーションに対するCPU負荷/メモリ使用量は全く問題になっていない。

3. EPICSの一部導入

3.1 概略

データチャンネルプロトコルによる通信は単純であるため、使用方法が非常に簡単であるという利点

はあるもののサーバーとの通信が何らかのトラブルで切断された後の再接続やイベント通知などの機能は各プログラム開発者が実装する必要がある。もちろんこれらの機能をライブラリとして実装して、DChプロトコルを多機能にしていくという方針もあると考えられるが全体が複雑になってしまう恐れもある。そこでDCh自体は現状のまま使用し、多機能を求める部分に関しては既に多くの研究機関での実績があるEPICSを導入してその利点を生かす方針をとった。また、PFではLinacからのトップアップ運転の計画もあり、この場合にはLinac/KEKBとの通信も重要になるのでこの目的にもEPICS導入は適している。

まず最初にDCh->EPICSゲートウェイを開発し、現在ではPFの運転パラメータをKEKB/Linac制御ネットワークからも参照することが可能となっている。現状では読み込みアクセスのみで、制御には使用していないが、トップアップ入射時には使用する予定である。

3.2 Input/Output Controller (IOC)

従来、EPICSのIOC上で動作する主要部分(iocCore)はWindriver社の商用リアルタイムOSであるVxWorksTMに依存した部分が多くあった。しかしながらバージョン3.14以降はiocCoreをポータブルにする努力が続けられた結果、現在ではGNU/Linux, Solaris, Windows NT/2000/XP, RTEMS 等、様々なOS上で動作するようになった。この利点は非常に大きく、PFでの制御システム構築にあたっては、ハードなリアルタイム性が不要な部分に関しては安価なPCやVME-SBCでも商用ではないOSを使用して構築することとした。Linuxを利用する利点として開発環境が安価に構築できることの他に、リアルタイムOSに較べてソフトウェアのデバッグが容易という利点がある(クロスコンパイルではなく、セルフコンパイルでの開発が可能)。ただし、RF、マグネットのCAMAC制御部分に関しては安定性も考慮して既に長時間運用の経験があるPowerPC + VxWorks の構成とした。

今回の改造ではCAMAC等のフィールドバスは置き換えずにそのまま使用し、前節で述べたデバイスサーバをEPICSのIOCで置き換える。そのため、過去の制御ソフトウェアは使用できなくなり、制御タスクやGUIの部分も全て書き換えることとなる。しかし、制御プログラムはPF-ARやKEKBから流用できるものもあり、SAD^[5]スクリプトも使用できること、各種のGUI作成ツール(dm2k, edm, Python/Tk等)も使用可能なことを考えると運転やマシンスタディの利便性に大きく貢献すると考えられる。

改造後に使用する主なIOCの数は以下の通り

数量	CPU	OS	boot
3台	VME-PowerPC750	VxWorks	diskless
4台	VME-PentiumIII	Linux	diskless
7台	PC	Linux	HDD

通常のLinux-PCはLAN/GPIB経由でのGPIB制御や、Ethernetベースの機器(マルチメータ、PLC、オシロスコープなど)の制御に使用するほか、後述のArchiverとしても使用する。

3.3 アーカイブ

データ保存・復元ツールとしてはいくつかの候補があるが、近年EPICSツールキットの中でも幅広く使用されるようになったChannel Archiver^[6]を採用することにした。

ArchiveEngineプロセスはEPICSのチャンネルアクセス(CA)プロトコルでデータを収集し、ディスクに保存する。レコード名やモニター方式(periodicalまたはmonitor)、ディスクへの保存間隔、ログ、ファイル切り替えなどの定義ファイルはXMLで記述する。定義ファイルを生成するためのデータは他の制御プログラム同様にバージョン管理システムのひとつであるCVS(Concurrent Versions System)を使用して管理し、スクリプトによる生成をおこなっており、今のところリレーショナルデータベースの導入は考えていない。

制御ネットワーク内では通常のパーソナルコンピュータ上でArchiveEngineを動かしている。ローカルディスクに保存されたデータはrsyncコマンドをcronで定期的に実行することによって所内ネットワーク側に置かれたWebサーバーに送られる。両方のネットワークに足を出したNAS(Network Attached Storage)を利用する案もあったが、データ二重化の意味もあるため現在の方式とした。

以前のバージョン(version 1.9台)ではCGIインターフェースが標準でサポートされていたが、version 2以降は廃止されてXML-RPCデータサーバ経由による取り出しが標準となった。このプロトコル対応のjavaクライアント等も存在するが、ユーザーの利便性を考えてデータを取り出すCGIスクリプトを作成し、Webサーバー上で動かしている。これにより通常のWebブラウザから全てのアーカイブデータをプロットすることが可能であり、取り出し速度も前バージョンとほぼ同じ速度が得られている。また、1日の電流、ビーム寿命履歴などの定型フォーマットのデータは研究所外からもアクセス可能なWebサーバーにも転送されている。

既に1年間にわたって真空のパラメータや、データチャンネル経由で送られてきたビーム電流履歴などはChannel Archiverで保存されている。1日あたりの保存データ量はばらつきがあるものの、真空関連のみで80~130MB/day程度となっている。全てのグループのデータを保存した場合でも1日あたり600MB以下になる見込みであるのでこの程度は問題ないと考えている。また、NAS等を有効利用すれば安価に長期間のデータを保存することは容易である。

4. VME-CPU割り込み測定

マグネット小型電源制御にリアルタイムOSを使用

せず、通常のLinuxオペレーティングシステム+EPICSの環境でどの程度の応答が得られるかを測定した。CPUにはGE-Fanuc社製のPentiumIII CPUを使用し、運転上の利便性と安定性を考えてディスクレスブート構成とした。このボードはIntelのネットワークチップを使用しており、PXE(Preboot eXecution Environment)を利用したネットワークブートが可能である。また、Linuxカーネルの再構築を行い、ルートディスクをNFSマウント可能にすることと、clock tickをデフォルト値である10msから1msに変更した点が通常のLinuxとの差である。試験にはINTERRUPT REGISTER(林栄精器,RPV130)を使用し、外部からのパルス入力でVMEバス割り込み→EPICSの割り込みトリガー→パルス出力、というソフトウェア構成にし、入出力のパルスジッタを測定した。短時間の応答を図3に示す。Linuxの場合、VxWorksに比べて応答のジッタは大きい。測定の間にflood pingを送るなど、負荷をかけて長時間運用する測定も行ったが、最大でも1ms以内の応答は得られている。ハードリアルタイムが不要の場合はこの程度の応答で十分であろう。

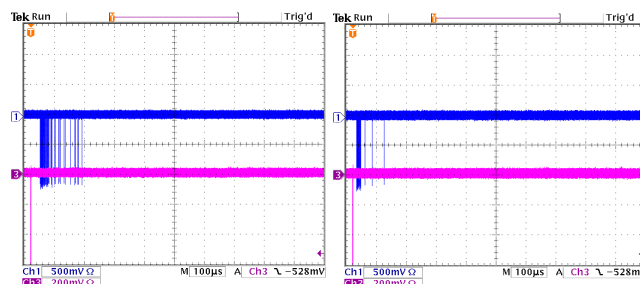


図3：VME-CPU割り込み測定
(左:PowerPC+VxWorks, 右:Pentium+Linux)両方とも下のトレースが入力トリガ、上が出力パルス(水平軸: 100 μ s/div)。

5. 謝辞

EPICS導入にあたっては加速器研究施設の山本昇様をはじめ、KEKB制御グループの方々、Linac制御グループの方々に大変お世話になりました。感謝します。

参考文献

- [1] 谷本育律、他 "PFリング直線部増強のための真空系更新"、In these proceedings
- [2] <http://www.aps.anl.gov/epics>
- [3] Photon Factory Activity Report (1998) p.120
- [4] Photon Factory Activity Report (1996) p. A-12
- [5] SAD Homepage <http://acc-physics.kek.jp/SAD/sad.html>
- [6] <http://ics-web1.sns.ornl.gov/~kasemir/archive>