RELIABILITY CONSIDERATION OF ACCELERATOR CONTROLS AT KEKB AND LINAC

Kazuro Furukawa*[A)], Atsuyoshi Akiyama[A)], Eiichi Kadokura[A)], Artem Kazakov[A)], Ichitaka Komada[A)], Katsuhiko Mikawa[A)], Tatsuro Nakamura[A)], Masanori Satoh[A)], Tsuyoshi Suwada[A)], Tomohiro Aoyama[B)], Shiro Kusano[B)], Takuya Kudou[B)], Yoshikazu Mizukawa[B)], Takuya Nakamura[B)], Kenji Yoshii[B)]
[A)]High Energy Accelerator Research Organization (KEK), 1–1 Oho, Tsukuba, Ibaraki, 305-0801
[B)]Mitsubishi Electric System and Service, 2-8-8 Umezono, Tsukuba, Ibaraki, 305-0045

Abstract

KEKB linac and ring have been operated for about ten years and their control systems have evolved during the operation to meet the required functionalities. Because of the nature of those factory machines, both quality and availability of the beams are important. Thus, we have always paid attention to the reliability of the accelerator control system. Here we'd like to examine the basic technical items to maintain the reliability; surveillance, testing frameworks, redundancy. These would be also important in future accelerators.

ＫＥＫＢと電子入射器の制御システムと信頼性

1． はじめに

KEKB 電子入射器と非対称衝突リングでは実験開始から約 10 年が経過し、さまざまな性能向上の要求に応えるためにその制御システムについても改善が進められてきた [1, 2]。そのファクトリ加速器であるという性格から、性能と可用性の双方が特に求められている。従って、これらの加速器制御システムの信頼性の維持には常に注意を払う必要がある [3]。以下では、まず、KEKB の入射器と蓄積リングの制御システムの概要を述べ、さらに監視、試験機構、冗長性、といった観点から信頼性を維持するための基本的な仕組みについて考えてみたい。

2． KEKB 運転制御システム

KEKB の入射器と蓄積リングの制御システムは、双方とも前の TRISTAN プロジェクトから部分的に資源を引き継ぎながら、KEKB プロジェクトでの要求を満たすための新しい機構の導入が行われた。

2.1 入射器

電子入射器 (Linac) の制御システムは 1982 年から 10 年以上の運転を行った後、老朽化のために 1993 年主要部分の更新が行われた。約 3 年の準備期間を経ていたが、KEKB 計画が認められることも予想して長期間利用可能であることに主眼を置き、業界標準または国際標準をできるだけ採用し、通信ネットワークとしては安価なフィールドネットワークは避け、イーサネット・TCP/IP だけのネットワークとした。ただし、KEKB 建設期においても PF 放射光施設への入射のために長期の停止期間がなかったために、旧システムの制御機器などの資源を 5 年以上かけて徐々に置き換えていくことになった。

ソフトウェアは準備期間にほとんどが書き換えられ、装置ハードウェアを表す制御装置表現層とビーム運転ソフトウェアで利用される加速器表現層の 2 階層で構成された。さらにその上に運転アプリケーションプログラム層がある。その後、KEKB 衝突リングの EPICS システムとの間にはいくつかのゲートウェイソフトウェアが設置され、さらに新しい部分については EPICS のソフトウェアを導入することを基本としているが、15 年以上経ったソフトウェアも基本的には同じ制御システム内で運用されている。

2.2 蓄積衝突リング

リングの建設の 5 年間にはビーム運転の必要がなかったため、制御システムについても、以前の TRISTAN プロジェクトで使用されたものの入れ替えが行われた。特にソフトウェア環境については EPICS を採用し、国際協力で開発されている利点を生かすことができた [4]。

ハードウェアについては TRISTAN での CAMAC の資源も約 40 台利用されているが、測定用の約 200 台の VXI インターフェース、多数の制御装置の接続に用いられる約 200 セグメントの ArcNet フィールドネットワーク、一部の測定器に GPIB、インターロックに PLC、など各装置に適した入出力装置が導入されている。実時間処理には約 100 台の VME 計算機、操作表示やビーム運転ソフトウェアの処理には Unix 計算機が多数用いられている。

Linac と KEKB の双方で、スクリプト言語の利用は顕著で、Tcl/Tk、Python/Tk、SADscript/Tk が幅広く運転に利用されている [5]。

3． 信頼性

加速器の運用を行うにあたって、信頼性の確保だけでなく、さまざまな場面で相反する要求の迫られることがある。理想的な解決と現実的な解決がかけ離れてしまう場合もあり得る。また、対象と考える部分については柔軟性を求めるが、他の部分に対しては確実性を求めることになりがちである。

*E-mail: <kazuro.furukawa@kek.jp>

時間、資源、安全性、保守性の制約とバランスを考えながら、解決法を探す必要がある。つまり、現実に適応した信頼性の維持を考慮することになる。

3.1 信頼性の実現

ここでは、次のような信頼性の実現について考える。

- まずは理想的な方法を探る、最も正しいと思われる選択を用意し他の方法と比較できるようにする。
- 実装されている機能について正しく運用されているかどうかすべて監視する。過去に起こった障害を繰り返さないための仕組みでもある。
- さまざまに用意される試験機構を系統立てて使えるように準備し、適当な場面で試験を行う。
- 必要な場合は冗長機構を採用して可用性を向上させる。高価にはなるが利点も多い。

3.2 より正しい理想的な方法

まずは理想的な基本の解決法を検討して、方向性を見失わないようにすることが重要だと思われる。例えば、過去の経験から新しい技術はより信頼性が高く設計されていることが多い。将来とも生き残る技術かどうかの見極めは必要であるが、新しい技術の何が優れているのか把握しておくことは重要である。

もしも加速器全体を表現することができるクラスオブジェクトがあれば、個々の機能の実装での新規開発は最低限に抑えることができるかもしれない。例えば、よく考えられた名前付けルールだけでも誤りを減らすためには有用である。

3.3 監視機構の充実

一度動作させることができた制御機構は将来とも動作することを期待したい。また、なんらかの誤りのために起こった障害を繰り返したくはないものである。

制御運転システムにはたくさんのソフトウェアが含まれているが、特に運転ソフトウェアの一部には残念ながら開発時にさまざまな仮定をしていて、その環境でしか動作しないものもある。そのようなソフトウェアを誤った環境で起動をすれば、誤った動作をすることになる。また、さまざまな目的のために当初の想定を超えて計算機やネットワーク機器を設置している場合もあり、それらの設定の管理が充分に行き届かなければ、誤った設定が放置され障害につながる可能性がある。単にハードウェアの寿命が尽きたことが見過ごされる場合もある。

従って、これらのソフトウェアやハードウェアの機能の内、何が不可欠な機能であるかを再確認することが重要である。それらの機能を運転中に試験する方法を見つける必要がある。試験を自動的に繰り返し、もしも制御機能に問題が見つかったら、記録を残し、運転に障害を与える前にアラーム機構や、電子メールによってその機能の管理者に連絡する必要がある。その制御機能によってはそれを自動的に再起動してしまうことが好ましいものもある。また、加速器の運転者への自動報告が必要なものもある。

このような監視機構は後追いの機構であり、事前に障害の対処ができるような準備をするのが理想であるが、限られた人的物的資源のもとでは現実的な対応策として有効であることが多い。

3.4 試験機構の充実

さらに、加速器の運転性能の向上のためにソフトウェアやハードウェアの運用環境を変更する場合があるが、ある機能が不完全にしか仕様を満足していなかったために、環境の変更が問題を表面化させ障害に結びつくことは多い。

もちろん必要最低限の試験は行われているはずであるが、上のような事態に備えるために、仕様に対するできるだけ完全な自動化した試験を準備しいつでも確認できるようにしておくことが望ましい。例えば、ソフトウェアについて、最近のオープンソースソフトウェアはさまざまに異なった環境で利用されるため、自動化された試験機構を準備している場合が多く、我々の制御機能についても同様な機構は有用である。

試験を自動化するための共通の枠組みを用意することが好ましいが、それは以下に述べるようないくつかの異なった試験に対応している必要がある。

まずは、既存の試験機構を系統的に整理し、過去に行われた試験を繰り返し行えるように整備し、環境が変わるたびに試験が行えるようにしておくことが望ましい。

最も単純な試験はユニット試験と呼ばれ、基本的な機能について独立に一つずつ順番に試験を行い、動作を確認する。例えば、EPICS 制御ソフトウェアでも runtests と呼ぶ試験の枠組みを用意しており、これまでに用意されてきた基本試験プログラムを自動的に実行することができる。また、利用者も必要な試験項目を追加することができる。

個々の基本的な機能には問題はないが、複合的な機能に障害がある場合には、ユニット試験では問題を発見できないので、網羅的な複合試験を行う必要がある。実際には効果的な複合試験を構築するのは容易ではないが、典型的な情報を実際に運転で使用するソフトウェアに与えてみるところから始めるといいのではないかと思われる。

さらに、ネットワーク機器や計算機などの設定が正しくない場合に、極端な条件のもとでは正しく動作しない場合もあり、負荷試験を行うことがそのような問題の発見に有効である場合がある。問題が見つかれば、設定を見直したり、すぐに修復できない場合は少なくとも何らか復旧の手順を再確認しておくことが望ましい。

加速器の保守停止期間が終わり、運転の始まる直前になんらかの問題が見つかる場合も多い。このような時期にはソフトウェア機能やハードウェアが一つずつ立ち上がってくるので、試験を効果的に行うことが難しい。停止期間中には新しいソフトウェアやハードウェアが導入設置されているので、元の機能を回復することも保証されているわけではない。また、停電などもあり、一年

前に行った変更が次の停止期間後に障害を引き起こす場合もある。

そこで、運転開始前にすべてのソフトウェアやハードウェアの制御機構の動作を確認するような巧妙な試験機構を用意することが望ましい。現在、KEKB や Linac ではこのような自動試験機構はなく、過去の経験から作成されたリストを元に人間が制御機能の確認を行っているが、少なくとも一部は容易に自動化することができると思われる。

3.5 冗長性の導入

冗長機構の導入は試験などまで考えると場合によっては負担が重く、信頼性の向上においては最後に考えることかもしれない。それにも関わらず障害の回復だけでなく、保守時にもおいても機能を失わないという意味で非常に有用であることが多い。

重要な制御機構についてはいずれにせよ別のソフトウェアやハードウェアを予備として用意しなければならないので、切り替え機構は大きな投資ではない場合もある。冗長機構が有用である事項は多く、確立した技術も多い。

現在、RAID やミラーディスクなどは民生用にも広く使われており、これらは冗長機構が現実に有効に機能し確立している例である。また、ネットワークファイルサーバもその重要性から冗長性が有効となる場合も多い。KEKB と Linac においても 10 年以上前からいくつかの冗長ファイルサーバを使用してきた。複雑であるため設定が十分でなく、期待した動作をしなかった場合もあったが、夜間の運転中など助けられた場合もあった。さらに、保守時にファイルサーバ機能を止める必要がなくなるので、さまざまな他の機能の保守を行う際にスケジュールの制約がなくなることは、保守期間の時間的な制約を考えるとたいへん有効である。

ネットワークシステムにおける冗長機構も確立された技術の一つである。KEKB 計画の以前にも冗長イーサネットトランシーバという比較的単純な機器を用いて、40 以上の光イーサネット接続を冗長化し、障害が起こった場合にも期待したような切り替えによって運転の停止に回避した事例がいくつかあった。現在は技術がさらに確立し、ラピッドスパニングツリーや HSRP (VRRP) と呼ばれるプロトコルにより、スイッチ、ルータ及び機器間接続の高可用性が実現できるようになっており、KEKB や Linac での制御の信頼性の向上に役立っている。

最近は比較的遅い制御の実装に PLC (Programmable Logic Controller) が使用されることが多く、Linac においても 150 台ほどが使用されているが、PLC の冗長化も実用性が高まっている。高価ではあるが CERN などでは利用も始まっており、安全に関わる部分などでは応用を考えてもいい時期に来ていると思われる。PLC の構成要素として電源、CPU、ネットワーク接続、バックプレーン、さらにはハードウェア側の配線の工夫により入出力まで冗長化できる可能性があり、用途によってが採用が可能かと思われる。

EPICS の冗長化 IOC (Input Output Controller) は DESY との共同研究として開発を進めている。この機構にはそれぞれの重要機能の正常性を常に監視する RMT (Redundancy Monitoring Task) と、2 台の IOC の間の内部状態と変数の同期を常に維持する CCE (Continuous Control Executive) が中心の機構として実装されている。RMT が監視する重要機能は RRR (Primary Redundancy Resource) と呼ばれ、スキャンタスク、チャンネルアクセスサーバタスク、シーケンサタスク、ドライバタスク、などがあるが、元の実装に変更を加え RMT と通信ができるようになっている。

KEK では EPICS ゲートウェイとして用いている IOC の重要性が高まっているので、その IOC を冗長化するために、DESY では実時間 OS (VxWorks) だけを対象としていたが、OS 独立性機構 (OSI) という仕組みを導入し、冗長化 IOC を使用できる環境を一般化することに成功している。

信頼性の向上に重点をおいたモジュラ計算機システムであるアドバンスト TCA (ATCA) の利用が ILC 加速器においても検討されているが、将来は冗長化 IOC を ATCA でも実行できるように開発を検討している。

以前から Linac の制御サーバは冗長化するように努力しており、しばしば有効であるが、他にもソフトウェアの冗長化が有効である部分は多いのではないかと思われる。

4. まとめ

KEKB と Linac においては、EPICS と SADscript が大きな成功をもたらした。また、Linac においては、古いソフトウェアと EPICS の共存もはかられており、それぞれが目的別に使用されている。KEKB のコミッショニングが始まってからほぼ 10 年を経過して、運転用アプリケーションソフトウェア開発は日々続いているが、内部機能については信頼性の維持がむしろ話題になってきている。ここでは信頼性の維持のための検討事項を挙げてみた。これまでに実装されてたものは多くはないが、それぞれ効果を発揮しており、将来の加速器においても重要となると思われるので、検討を続けていきたいと考えている。

参考文献

[1] K. Furukawa *et al.*, “Accelerator Controls in KEKB Linac Commissioning”, *Proc. of ICALEPCS99*, Trieste, Italy, 1999, p. 98.

[2] N. Akasaka *et al.*, “KEKB Accelerator Control System”, Nucl. Instrum. Meth. **A 499**, 2003, p. 138.

[3] K. Furukawa, “Modern Accelerator Control Systems”, *Proc. PAC2007*, Albuquerque, US, 2007.

[4] `<http://www.aps.anl.gov/epics/>`.

[5] `<http://acc-physics.kek.jp/SAD/sad.html>`.

[6] K. Furukawa *et al.*, “Lifespan of an Accelerator Control System and Transition to EPICS”, *Proc. of 2nd Annual Meeting of Particle Accelerator Society*, Tosu, 2005, p.424.