

J-PARC MR MPS 装置の紹介とそこから派生されることの提案

THE INTRODUCTION OF THE J-PARC MR MPS EQUIPMENT, AND THE DISCUSSION OF THE INTEREST CONCERNING THE EQUIPMENT

中川 秀利[#], 秋山 篤美, 佐々木 信哉, 木村 琢郎

Hidetoshi Nakagawa, Atsuyoshi Akiyama, Shinya Sasaki, Takuro Kimura

KEK

Abstract

There was not large trouble on the operation, after the MPS of MR of J-PARC was constructed. However, the addition of the equipment is not possible in the problem of the supply system of IC, and such board has carried out the revision. The revision can be freely carried out, since it is done to planning, design, manufacturing request in selves. The revision of CPU board of the MR-MPS system was undertaken, because there was the possibility of the expansion of the MPS of MR. We consider that the equipment which can be used for the operation was completed. Though, the problem like MAC-address was found. Revised CPU board is explained, and it is discussed that staffs choose the mind.

1. はじめに

KEK と JAEA で協力して運用されている J-PARC は 3 GeV の陽子ビームで中性子やミュー粒子を発生させて実験をしたり、30 GeV の陽子ビームでニュートリノや K 粒子を発生させて実験を行う複合施設である。世界最大のビーム強度の実現を目指して運用と改造を行っている。そこで、機器異常時などには十分短時間のうちに対処できる機能が要求される。それを実現したのが Machine Protection System(MPS) である。

J-PARC 派の Linac、RCS、MR の 3 段の加速器から構成されるが、それぞれ時間的特性が異なる。Linac、RCS は繰り返しが 40 ミリ秒で同期している。そこで統一した MPS 装置が使われた。MR は繰り返し時間が計画当初で 3.64 秒、運転時には 2.48 秒から 6 秒と長い。しかも、Linac、RCS とは異なる機能も要求される。上流からのビームを止めるには当然だが、MR 内にあるビームをアポートダンプに捨てることも要求される。この様に 2 つのグループで要求が異なるので異なる機器が作られることになった。

MR では頻繁に変わる複雑な要求が発生することが想定されたので、その要求に対応できるだけの MPS 装置が作られた。その基本性能は現時点でも通用するし、故障もしないのだが、追加するための IC の供給などに関して不安が出てきたので、各モジュールを新しい IC に置き換えたものへと改版を進めている。その中でも難しいのが CPU ボードであるが、試作品を機能させることができるようになった。これを中心に J-PARC MR の MPS 装置を紹介するとともに、この作業の結果の拡張の話を出席者と相談し、また、自作ボード作成時の問題点の相談をしたいと考えている。

[#]hidetoshi.nakagawa@kek.jp

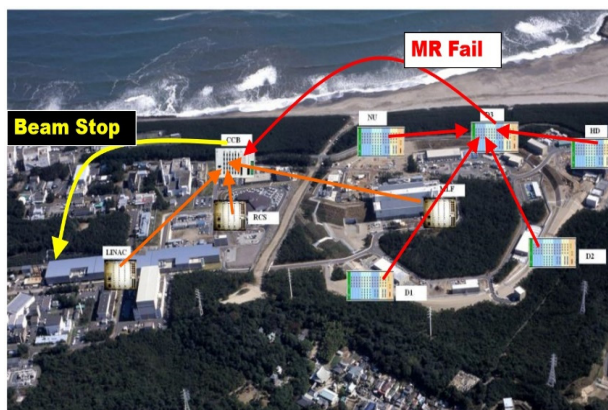


Figure 1: MPS signal flow of J-PARC.

2. MR の MPS 装置の概要

J-PARC の全体の様子とその配置されている MPS 装置の様子を Figure 1 に、MR の機器配置を Figure 2 に示す。Figure 2 に示されているように MR の MPS は種類の機器グループからなる。一つは『MR-MPS』と表記される、機器グループからの機器異常信号を受けて論理処理をした後で次の加速のビームを止め、また、リング内のビームを廃棄する指示を出すシステムである。もう一つのグループがある。『BLM-MPS』と表記されてビーム損失モニタが閾値を超えるビーム損失を検出したときに『MR-MPS』へビーム異常があったことを集約し伝達する装置である。検出器が約 300 本あり更に各信号の積算量と瞬時値の両方を信号源とするために、『BLM-MPS』では入力点数を多くし複雑なことはしないことにした。

ここでは『MR-MPS』の CPU ボードに関する議論に限定する。このボードは開発途上の加速器と言

う、運用条件の変わりやすい機器構成に柔軟かつ迅速に対応できるように設計されている。

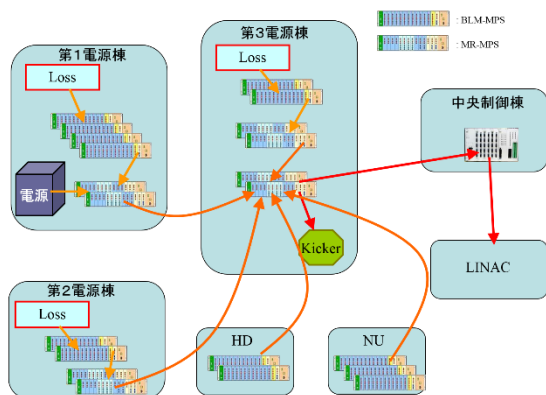


Figure 2: MPS about MR.

また、入出力の機器が幾つかの接続方式をとるために、入出力をモジュール化し、構成を柔軟に変更できる。J-PARCで作ったモジュールを Table 1 に示す。

Table 1: Modules

ID	Name	Use	Input	Output
1	MR-MPS-FCT	Optical transceiver	2	2
2	MR-MPS-OPT	Relay signal input	4	0
3	MR-MPS-OLC	Optical signal input	4	0
4	MR-MPS-TTL	TTL signal input	4	0
5	MR-TTL-GIO(TTL)	TTL In/Out	2	2
6	MR-TTL-GIO(Relay)	Relay In/Out	2	2
10	MR-MPS-CPU	CPU(I/O)	0	0
11	MR-MPS-POWER	Power Supply	0	0

3. 新 CPU ボード

MR の MPS の最初の CPU ボードには SUZAKU と呼ばれる商品を論理素子として、また、CPU として使った。CPU が組み込まれた Xilinx 社の Virtex-4 を使っている。CPU が FPGA 内に組み込まれているので、外付けのような配線が不要で信頼性が高いと判断し、この方式を採用した。

Table 2: Information of FPGA/CPU

Maker	ALTERA
Product	Cyclone V 5CSXFC6D6F31C8N
CPU	ARMv7 Processor Brought up 2 CPUs
Memory	1 GB
OS	Linux version 3.10.31-ltsi-05131-g55fd0e (Yocto Project)

新しい CPU 基盤もこの概念を踏襲した。しかし、

FPGA の乗った子ボードを採用するのではなく、CPU ボードに直接 FPGA を取り付けた設計とした。今回採用したのは ALTERA 社の Cyclone V で CPU として ARM が採用されている。これらを Table 2 にまとめる。

論理素子数が増えるので、機能拡張が可能だと考えているが、現在のところ、旧 CPU と同じ条件で、同じ環境下で同等に見えるようにしている。^[1]

4. 新 CPU ボードの開発状況

4.1 Version 1

開発練習のため試作を行い、何が起こるか、どのような問題が発生するか確認した。配線ミスなどのケアレスミスと、高速の電子回路であることによる重大な製造上の問題点など貴重な情報が得られた。動作は不安定で実用には使えないが、試作 2 号機では実用になる製品が作れそうな確認はできた。

4.2 Version 2

試作 2 号機は十分な動作試験に耐えられるものを目指して、入念に再設計を行い、詳細な指示を行って発注を行った。その結果、本質的ではない問題点が残ってしまったが、現在の試験まででは十分に運転に耐えられるものができた。新 CPU ボードの写真を Figure 3 に示す。

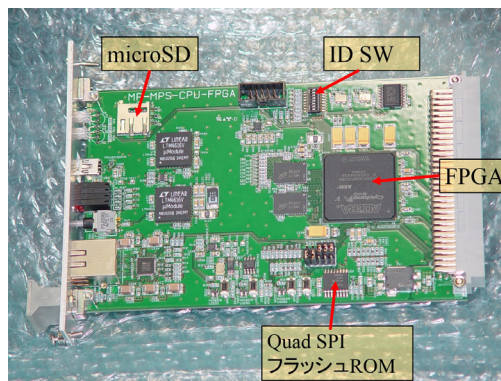


Figure 3: Photo of new CPU.

J-PARC の加速器の MR グループで使っている MR-MPS 装置は Figure 2 にあるように 8 セットであるが、これらの旧 CPU ボードを新 CPU ボードに置き換えて動作確認を行い、1 日程度の監視下では異常なく機能することを確認した。

2015 年 8 月中旬からは、2 セットで連続的な動作試験を行う。異常がなければ、旧 CPU の経年劣化で異常が発生する前に交換する予定である。

4.3 CPU ボードの使用方法

最初の立ち上げは microSD カードからプログラムをロードする。ロード時間の都合で Class 10 が

要求される。一度立ち上がるとフラッシュメモリーに書き込まれ電源投入時にはフラッシュメモリーから立ち上がり、microSD カードは要らなくなる。

各サブブラックの識別のために CPU ボードに ID SW がつけてある。この番号に基づいた EPICS の立ち上げプログラムが使われる。CPU ボード交換時は ID SW の値を合わせることで、同一機能を実行させることができる。また、監視の都合上それぞれの場所とネットワーク上の名前<IP-Address>が対応しているのが望ましいので、ID をもとに IP-Address も変更するようにしている。^[2]

あとは、NFS で接続されるサーバ上の EPICS の立ち上げプログラムを管理すると MPS 装置を適切に機能させることができる。これは RDB 管理下で判りやすい作業で済むように構成されている。^[3]

5. 解決すべき問題

自主開発で、単純に商品として FPGA を買って使ったので、イーサネットを使うときに MAC-address が決まっていない。試験段階では問題にならないが実用段階では無視できない問題である。

J-PARC の現在の担当者はまだ方式を決定していない。それで思いついた解決方法が『加速器学会で聞く』であり、このトークになった。

このプロシーディングス読んでくださった方は中川まで意見を下さい。

自主開発で MAC-address はどうしているか？

- ① 個別に IEEE に申請
- ② 自分で使っている領域を適宜使用
- ③ 研究所（事業所）内で総合的に管理（事業所で IEEE 申請）
- ④ 研究所（事業所）内で総合的に管理（事業所で IEEE せず）

また、この装置は KEK 内の他のグループで評価されたり使用されたりもしている。また、「海外の研究所で開発されたものを買って使う」という意見もよくある。この MPS 装置は汎用性があり、加速器の MPS に用途は限らない。信号はデジタルであるが、入力 40 本程度で 8 出力あり、その間で高度な論理処理ができる装置。論理処理時間は 100 ナノ秒以下。いろいろな用途があると思われるが、使用希望組織は出てこないか？

KEK (J-PARC) の MPS 装置を皆さん使いませんか？

- ① 安ければ使う。
- ② 稼働状況を見て考える
- ③ もっといいのを作る
- ④ 回路図もらって作る

くく 動かないです。

それからいろいろな組織で独自に同じようなのを作っていると推測しますが、皆さんはどのように開発をしておられるのでしょうか？

いろんなところで、MPS 装置のようなものを開発する時

- ① 自主開発
- ② 既製品の購入
- ③ 海外で発表されたもののコピー製造
- ④ 国内で発表されたもののコピー製造

この様な話をトークの時でも、メールでも、教えて頂けると助かります。

6. 最後に

ちょっと野心的な計画だが、メンバーが CPU の設計をできるようになった時、その能力は有効に生かせる環境が欲しい。例えば、ビーム電流を測定する DCCT をフルデジタル制御と測定とで、アナログ回路につきもののオフセットをなくす。情報はデジタルのまま分配する。

FPGA の高速論理処理と、CPU の EPICS 処理が一体化している製品を使いこなせるのは、ノイズ対策や構成の単純化とそれに基づくミスの低減などのメリットから、加速器制御に非常に望ましいと思う。

最後の最後に、先ほどの MAC-address の問題の解決に加速器学会で力を貸してもらえないでしょうか？

J-PARC MPS に限らず同等の問題抱えるところが増えると予想される。

加速器学会として IEEE に申請して、会員に 16 単位くらいで分ける

- ① YES
- ② NO

参考文献

- [1] H.Nakagawa, et al., “多重モード型 MPS(Machine Protection System)の実装方法”, Proceedings of the 7th Annual Meeting of Particle Accelerator Society of Japan, Himeji, August 4-6, 2010.
- [2] H.Nakagawa, et al., “J-PARC MR の MPS 装置の異常対策時の時間短縮と誤作業低減のための改造”, Proceedings of the 11th Annual Meeting of Particle Accelerator Society of Japan, Aomori, August 9-11, 2014.
- [3] H.Nakagawa, et al., “J-PARC(MR)の MPS のマネジメントの自動化”, Proceedings of the 8th Annual Meeting of Particle Accelerator Society of Japan, Tsukuba, August 1-3, 2011.