

Raspberry Pi を用いた EPICS 対応 DC マグネット電源用制御ソフトウェアの開発

DEVELOPMENT OF CONTROL SOFTWARE FOR EPICS COMPATIBLE DC MAGNET POWER SUPPLY USING RASPBERRY PI

草野史郎 *^{A)}、榎本嘉範^{B)}、牛本信二^{A)}

Shiro Kusano*^{A)}, Yoshinori Enomoto^{A)}, Shinji Ushimoto^{B)}

^{A)}Mitsubishi Electric System & Service Co., Ltd.

^{B)}High Energy Accelerator Research Organization (KEK)

Abstract

In order to fully support EPICS for controlling the DC magnet power supply, we have developed a new power supply that uses the Raspberry Pi 4 as the main controller. Linux is adopted as the OS of Raspberry Pi4, and EPICS IOC is executed on Linux. A 20-bit DAC and a 24-bit ADC are used to monitor and control the power supply. These are directly connected to the Raspberry Pi using the SPI. It also uses an Arduino mega to process signals such as interlocks and connects to a Raspberry Pi using a UART. This paper reports the details of software using EPICS.

1. はじめに

KEK 電子陽電子入射器 (以下入射器) では、Energy の異なる 5 つのリング (PF、PF-AR、SuperKEKB HER/LER、および陽電子ビーム用ダンピングリング) にビームを供給している。入射器では、約 550 台のマグネットを使用しており、それらを制御するシステムは、PLC の DAC/ADC モジュール、PLC および LXI 対応電源 [1] を用いた DC マグネットシステム、cRIO と PXI を用いたパルスマグネットシステムの 3 つのシステムで構成されている。そのうち約 200 台の DC ステアリングマグネットの電源制御は、出力制御はアナログ電圧制御、出力モニターはアナログ電圧出力でおこなわれている。それぞれが PLC の DAC、ADC モジュールに接続され、制御はラダープログラムで行われている。加速器の基幹制御システムは EPICS を使用しており、PLC と EPICS との間を変換するため外部の計算機上のソフトウェアを実行している。今回これらの電源の置き換えとして、電源内に Linux が動作可能なシングルボードコンピューターである Raspberry Pi [2] を搭載し、Linux 上の EPICS IOC (Input Output Controller) から直接 DAC/ADC を制御する電源 [3] とそのソフトウェアの開発をおこなった。本稿では、新 DC マグネット電源コントローラの制御システムについて詳述する。

2. 新 DC マグネット電源

入射器の DC マグネット制御システムは、電源出力およびインターロックは PLC を用いて制御をおこなっている。PLC は、長期間に渡って高い安定性を実現していたがネットワークの広帯域化や EPICS への対応などで機器の更新が困難になっている。近年、Raspberry Pi などのシングルボードコンピューターや Arduino Mega などのワンチップマイコンが安価になっている。これらの I/O を用いて新たなマグネッ

ト電源の開発をおこない、これまで EPICS での制御部を外部の計算機からおこなっていたものを、Linux を搭載した I/O を組み込むことで電源内部で EPICS に対応することが可能となった。

2.1 ハードウェア構成

Table 1 に、新制御システムで使用した主なハードウェアを示す。電源制御部としてシングルボードコンピューターである Raspberry Pi4 Model B を使用した。Raspberry Pi は、1.5 GHz、Quad Core の ARM プロセッサを搭載し、Linux ベースの Raspberry Pi OS で動作する。コントローラ内で使用するインターロック用接点や電源内の温度などアナログおよびデジタル I/O は、Raspberry Pi の I/O だけでは不足するため、ワンチップマイコンの Arduino Mega [4] を使用している。出力制御は、アナログ電圧で制御するため 20 bit DAC デバイスである Analog Device 社製の AD5791 (20 bit) [5]、出力モニターは 24 bit ADC で Analog Device 社製の LT2380-24 (24 bit) [6] を使用している。Figure 1 に電源内部の写真を示す。



Figure 1: Magnet Power Supply.

2.2 ソフトウェア構成

入射器では、EPICS を中心とした制御システムが構築されており、様々な機器が Channel Access プロト

* skusa@post.kek.jp

Table 1: Hardware Configuration

制御部	Raspberry Pi4 Model B
Interlock	Arduino Mega
DAC	Analog Device AD5791
ADC	Analog Device LTC2380-24

コルを介して制御されている。開発したコントローラの制御部では、Raspberry Pi を EPICS IOC として使用する。Table 2 に EPICS の環境を示す。

Table 2: EPICS Environment for Raspberry Pi

OS, Base, Modules	version
Raspberry Pi OS	2019.9.26 Ver
base	R3.15.17
asyn	4-38
drvAsynSPI	none
devgpio	R1-0-6
Sequenser	2.2.7
StreamDevice	2-7-7

2.2.1 GPIO 制御

開発したコントローラでは、Raspberry Pi の GPIO を使用する。EPICS レコードから直接 GPIO の制御をおこなうため、devgpio を利用している。このドライバサポートは仮想ディレクトリ配下のファイル"/sys/class/gpio/GPIO(xx)/Value"へ読み書きをおこなうことで GPIO の制御をおこなう。レコードタイプは、bi/bo をサポートしており、EPICS データベース内で DTYP("devGpio") を選択、IN/OUT フィールドにピン番号および値 (H, L) を指定して使用する。

2.2.2 SPI 通信

DAC および ADC デバイスの制御は、SPI 通信を介しておこなっている。DAC は、SPI0、ADC は SPI6 を使用している。SPI0 は、Raspberry Pi の初期設定で使用可能であるが、SPI6 については Raspberry Pi の設定ファイルである"/boot/config.txt"を書き換えて使用する。また、EPICS IOC から SPI 通信をおこなうための環境は、AsynDriver、Stream Device、drvAsynSPI を組み合わせて構築している。

2.2.3 シリアル通信

Raspberry Pi と Arduino 間は、当初 USB ケーブル 1 本で接続していたが、後述するトラブルにより UART 通信に変更している。EPICS IOC とのデータ送受信は、AsynDriver と Stream Device を用いた UART 経由のシリアル通信でおこなう。

3. 問題点

3.1 Raspberry Pi4 の GPIO

Raspberry Pi OS のインストール後、GPIO 試験時に GPIO コマンドが動作しないという問題があった。調べたところ、2020 年 5 月時点での Raspberry Pi OS の GPIO のドライバーである WiringPi(Ver.2.50) では Raspberry Pi4B の GPIO が動作しないことが判明した。最新の WiringPi(Ver.2.52) をダウンロードおよびインストールして GPIO コマンドが動作するようになった。

3.2 SD カードの内容破損

Raspberry Pi の電源を OFF/ON、または Raspberry Pi OS を再起動すると Kernel Panic が発生し正常に OS が動作しないことが度々発生した。SD カードへの書き込みと再起動のタイミングにより、SD カードの内容が破損されることが原因であった。この対策として、SD カードへの読み書きを極力減らす設定をおこなった。これにより、電源 OFF/ON、再起動しても Kernel Panic による動作不良が解消された。

- swap の無効化

swap は、使用していないメモリを一時的にファイルに退避する仕組みである。swap が頻繁に発生すると、その分メモリとファイル間アクセスが多くなる。swap を無効化することで、SD カードへのアクセスを減らし、SD カードの寿命も長くする効果も期待できる。

- syslog ソフトウェアの変更

syslog (System Logging Protocol) は、Unix OS の標準的なシステムログまたはイベントログなどを収集する機能である。Raspberry Pi OS の初期設定では、syslog はファイルシステム内に保存するソフトウェアがインストールされる。本コントローラでは、syslog から busybox-syslog に変更をおこなった。busybox-syslog は、ログをファイルシステムではなくメモリに展開する。このため、ファイルシステムへのアクセスを減らすことが可能になった。

- Raspberry Pi OS Boot 時のファイルチェック

Raspberry Pi の初期設定では、Boot 時にファイルシステムを検査、修復するコマンドである fsck が実行される。Raspberry Pi の Boot 時の振る舞いを決めるファイル"/boot/cmdline.txt"に"fsck.repair=yes"と記述されているところを"no"に変更をおこなった。この変更により、SD カードへのアクセスを抑制することが出来た。

3.3 電源出力の安定性

電源の出力安定度試験中に、出力電流が一定周期に大きく変動と約 250 μ A の 2 値化変動が観測された。Figure 2 に電源電流の読み返し値を示す。本図において、上段グラフは、初期状態の電流値を測定、中段グラフは Raspberry Pi の無線機器を OFF にした時の電流値を測定、下段のグラフは USB シリアル通信

の通信頻度を 1 秒から 0.5 秒に変更したときの電流値を測定したものである。縦軸は読み返し電流、横軸は時間を示している。

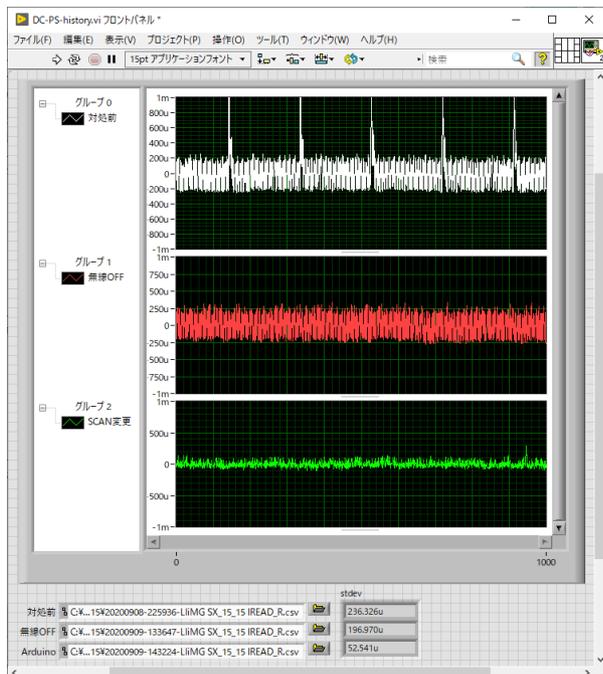


Figure 2: Stabilization test.

3.3.1 無線デバイスによる出力不安定

一定周期で電流が大きく変動する現象を Figure 2 上段グラフに示す。対策として、Raspberry Pi の設定で Bluetooth および Wifi の OFF をおこなった。これにより、一定周期の変動は解消された。

3.3.2 USB デバイスによる出力不安定

電源の出力安定性を測定時に、DAC 出力が 2 値化する現象が観測された。原因調査する中、USB シリアル通信を停止するとこの現象は発生しないことがわかった。この対策として、USB シリアル通信の間隔を 1 秒から 0.5 秒に狭め Raspberry Pi を常時高負荷状態に保つことで安定した出力にした。この結果、読み返し電流の変動幅が約 30 μA となった。量産化に向けた試験機において USB シリアル通信から UART シリアル通信への変更をおこなった。Figure 3 に UART シリアル通信時の出力の安定性を示す。通信間隔を変えて測定をおこない、上段グラフは、0.2 秒、中段グラフは 0.5 秒、下段グラフは 1 秒となっている。また、縦軸は読み返し電流、横軸は時間を示している。3 つのグラフの測定結果から、通信間隔を変えても読み返し電流の変動幅は約 17 μA と安定した出力が得られようになった。

3.4 32 bit ソフトウェアによる ADC 読み出し欠落

本コントローラで使用している 24 bit ADC デバイスは、読み出し情報として 24 bit の出力コードと 16 bit のサンプルレートなどの情報を合わせた 40 bit のデータを返す。しかしながら、Raspberry Pi OS は、32

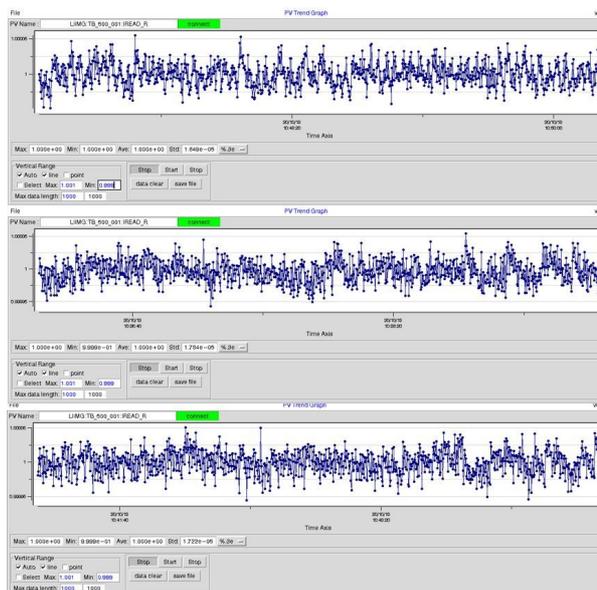


Figure 3: Stabilization test of UART.

bit ソフトウェアであるため、1 回の送受信ですべての情報を取得することが不可能であり、分割での情報取得も困難なことから、本システムでは 24 bit の出力コードのみをデータ取得する構成にしている。

4. まとめと今後の課題

入射器の電磁石電源システムにおいて、Raspberry Pi を利用した電源の開発をおこなった。2020 年の 9 月から試験運用をおこない問題無く動作することが確認できた。今後は、電源の量産化を進め多数の運用を予定している。今回は少数台であったため、Raspberry Pi OS などの設定が容易であった。今後は台数が増えることから設定および管理がより困難になることが予想される。Net Boot もしくは NFS root などのネットワークによるファイルの一元管理を検討し、試験をおこなう予定でいる。

参考文献

- [1] S. Kusano *et al.*, “KEK 入射器における EPICS EPID を利用した電磁石電源用フィードバック制御システム”, Proceedings of the 12th Annual Meeting of Particle Accelerator Society of Japan, Tsuruga, Japan, August 5-7, 2015
- [2] <https://www.raspberrypi.org/>
- [3] Y. Enomoto *et al.*, “Raspberry pi を用いた EPICS 対応 DC マグネット用パイポラ電源”, These proceedings
- [4] <https://www.arduino.cc/>
- [5] <https://www.analog.com/jp/products/ad5791.html>
- [6] <https://www.analog.com/jp/products/1tc2380-24.html>