EXTENDED MAINTENANCE AND GRADUAL REPLACEMENT OF MIL-1553-BASED FIELD NETWORK AT GSI

V. Kleipa*, R. Baer, M. Dziewiecki, K. Lueghausen, S. Rauch and M. Weiland
GSI Helmholtzzentrum für Schwerionenforschung, Darmstadt, Germany

Abstract

The control system of the GSI accelerator complex dates back to the early 1990s. It is based on a derivative of the MIL-1553 field bus, providing around 1 Mbit/s throughput and deterministic real-time operation. Although a modern control architecture with smart front-ends (System Control Units, SCUs), Ethernet, and White Rabbit is now widely deployed at GSI, several hundred devices still depend on the legacy network and protocols.

The ageing electronics and limited bandwidth provide strong motivation to migrate to the new control system. However, an upgrade is limited by various constraints: strict real-time operation, existing software architecture, physical space and available power sources. Therefore, operation of the legacy interfaces must be ensured for at least another decade while a unified migration strategy is hard to find.

In this contribution, we present a generic approach for extending the lifetime of the existing MIL-1553-based systems and gradual replacement of their components. This includes upgrades of bus controllers, device controllers, the bus itself as well as a new Modbus/TCP-based solution for simple devices.

INTRODUCTION

The GSI's legacy control system is still widely in use at various places in the accelerator complex, including the Unilac linear accelerator and partly the SIS-18 synchrotron and the ESR storage ring.

It uses a custom network solution, the *DeviceBus* [1] for connecting remote devices. DeviceBus is a serial, half-duplex, multipoint interconnect with single master and multiple slaves, net bandwidth of ca. 80 kB/s and range of ca. 200 m. Its physical layer is based on the MIL-Std-1553 bus, however major differences exist between both standards, including different connectors, different data model and lacking redundancy. DeviceBus is a hard real-time standard with short and deterministic latency of ca. 20 µs.

SYSTEM ARCHITECTURE

The concept of the system can be described as *central controllers approach*: relatively few control units (*SE crates*) communicate multiple devices over DeviceBus and *interface adapters* (IFA) as shown in Fig. 1.

The SE crates are quite complex, VME-based systems using multiple Motorola 68000 processors along with a main PowerPC-based computer. They are connected to the accelerator control network and the global timing system, which is another derivative of the MIL-Std-1553. They run the front-end software which communicates to higher control system layers via an in-house developed standard known as *Device Access* [2]. They constitute the boundary between the control software and the real-time hardware.

Interface adapters are in turn rather simple devices which react for incoming instructions on the fly. Therefore, the real-time operation of the system relies on deterministic response time of DeviceBus and interface adapters.

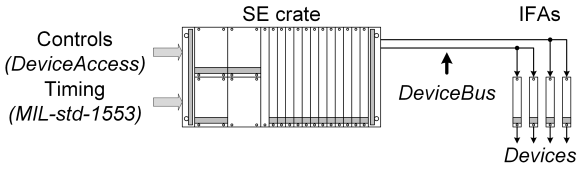

Figure 1: Legacy controls with SE.

This architecture stays in contrast to the new GSI's control hardware with *System Control Units* (SCUs) [3, 4], which can be described as *local controllers*. Here, every device is equipped with a fully-featured controller with access to the control network and timing (WhiteRabbit [5, 6]), and running the front-end software according to the newer FESA standard [7, 8]. There is no intermediate bus, which makes the architecture conceptually simpler — see Fig. 2.

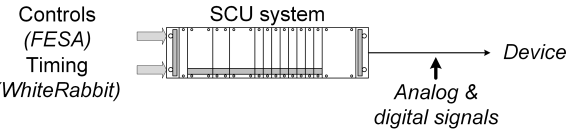

Figure 2: Direct device control with SCU.

UPGRADE STRATEGIES

Continuous upgrade measures tend to replace the legacy controls with SCU-based systems. However, direct replacement of interface adapters with SCUs is hardly ever possible without a deep upgrade to the controlled hardware itself: SCUs are physically larger, draw significantly more power (ca. 35 W vs. 4.5 W) and require better cooling than the interface adapters, and finally they have incompatible backplane interfaces.

Attempts undertaken to find a single replacement architecture for various use cases failed. Therefore, multiple solutions were needed to replace most critical parts of the old architecture, or to extend its lifetime if an upgrade is not viable.

An extensive description of the software upgrade effort can be found in [9]. Here we focus on selected hardware aspects of the upgrade.

* v.kleipa@gsi.de

Maintaining the Interface Adapters

Multiple versions of the interface adapters have been developed over the last 30 years, starting from discrete logic units through various programmable solutions (GALs, CPLDs) up to FPGA-based devices. The newest one, the IFA-11 (see Fig. 3) maintains high level of compatibility with all its predecessors and is additionally equipped with a 100 Mbit/s Ethernet connection.

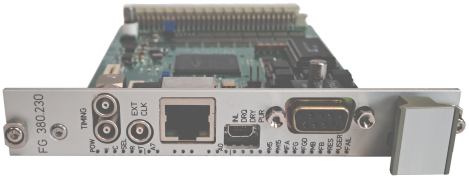

Figure 3: The IFA-11 interface adapter.

IFA-11 is built around an Altera MAX-10 FPGA. A versatile backplane interface allows connecting to various backplane standards used over years. A Cortex-M4-based microcontroller is used for interfacing Ethernet. DeviceBus data de-/serializing can be done either in a legacy way by a specialized chip (Intersil/Renesas HD-6408) or internally by FPGA firmware. An additional connector for piggy-boards allows future expansions.

A series of about 100 pieces of IFA-11 should act as a spare pool for existing systems and material for partial upgrades.

Replacement of SE Crates with SCUs

SE crates belong to the most critical parts to be replaced. An obvious reason is lack of spare parts. Also, they have insufficient computing resources for seamless implementing the FESA-based software. Therefore, they are being consequently replaced with SCU-based controllers.

To facilitate connecting the DeviceBus, a special extension card for SCU has been developed — the Serial I/O (SIO) [10].

The resulting architecture, shown in Fig. 4, does not require any hardware changes for the devices under control, and allows a migration to the newest FESA control software and WhiteRabbit timing.

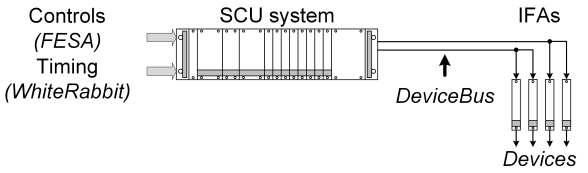

Figure 4: Controls with SCU and legacy bus.

A Faster Real-Time Alternative for DeviceBus

Bandwidth limit of the DeviceBus is getting increasingly problematic for ramped devices with simultaneous data acquisition. To address this problem, a next-generation interconnect has been developed, called *MIL-2*.

MIL-2 is a serial, fibre-optics-based link operating with an effective throughput of 2 MB/s. It provides hard real-time operation with latency of max. 2 µs and time resolution down to 32 ns. In contrast to DeviceBus, MIL-2 is a point-to-point standard: each device requires its own link to the controller, as shown in Fig. 5. Although this requires more cabling, bandwidth sharing is avoided and full-duplex communication becomes possible.

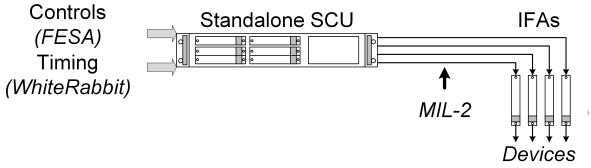

Figure 5: Controls with MIL-2.

A pair of multimode fibres (OM-1) with ST connectors is used as connection medium, along with Broadcom HFBR-1414 transmitters and AFBR-2418 integrated receivers. This configuration allows cable lengths of more than 1 km. The data are encoded using the 4b/5b + NRZI scheme into a 31.25 Mbit/s stream. The relatively low bitrate allows both generating and decoding the bit stream using cheaper FPGAs without dedicated transceiver hardware.

A specific feature of the DeviceBus is the presence of dedicated hardware lines for device status reporting and interrupts. In case of MIL-2, the same functionality is achieved by injecting the status information into the data stream. As MIL-2 is in fact a symmetric peer-to-peer standard, status signals are also transmitted from the controller to devices — these can be used as hardware triggers.

A complete transmitter chain consists of a status injector, a packet assembler including 4b/5b encoding, serializer and NRZI modulator. The receiver is quite symmetric to the transmitter, as shown in Fig. 6.

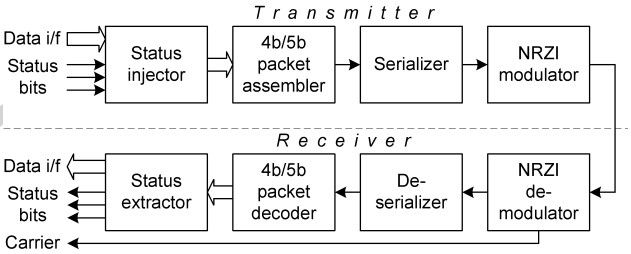

Figure 6: Data flow through MIL-2 interfaces.

An additional feature of MIL-2 is recovery of the carrier frequency from the incoming bit stream, which allows syntonizing the local clock of the interface adapter with the controller.

Maintaining the overall architecture of DeviceBus-based controls is a big advantage of MIL-2: upgrading the existing devices to the new standard does not need intensive software rework. Also, hardware measures are limited: on the controller side, the hardware is built using already existing control system components (SCU, fibre optic I/O cards). IFA-11 adapters with added piggy-board for optical communications are used on the interface side.

Ethernet-Based Solution

Ethernet is a simple, cost-effective solution for devices which do not need strict real-time operation. Some examples can be DC power supplies or air valves. In contrast to previously shown solutions requiring an SCU, an ordinary PC can be used as controller (*front-end server*), as shown in Fig. 7.

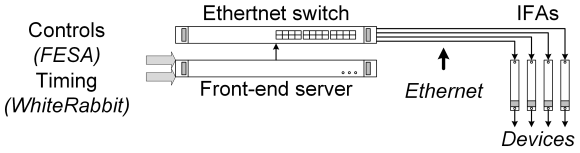

Figure 7: Ethernet as field bus.

Interface adapters form a local network with the controller. There are no hard constraints on the network hardware. Typically, the controller uses a 1 Gbit/s connection to a switch while interface adapters communicate with 100 Mbit/s. The local network is treated as a field bus and is isolated from the main accelerator control network. It uses static IP v4 addressing derived after legacy DeviceBus addresses of the interface adapters.

Current efforts aim at implementing the Modbus TCP/IP protocol [11] for device control. Further planning focuses on using the precision time protocol (PTP) to achieve real-time control for less demanding applications.

As IFA-11 is equipped with both DeviceBus and Ethernet ports, the transition between both standards can be carried out quite fluently.

CONCLUSION

A spectrum of solutions has been proposed and implemented to replace the components of the GSI's legacy control hardware. Their pros and cons have been shortly summarized in Table 1.

Table 1: Comparison of Various Upgrade Measures

Parameter	IFA Replacement	SE → SCU	MIL-2	Ethernet	SCU System
Data Throughput	-	+	+	+	++
Real-Time Operation	+	+	++	-	++
Compatibility	++	+	+	-	- -
Space Efficiency	+	+	+	+	- -
Affordability	+	-	+	++	- -

It is expected that careful selection of upgrade strategies will allow operation of legacy devices over their natural lifetime. This shall optimize costs and effort of the machine maintenance and give sufficient time for future major upgrades.

REFERENCES

[1] G. Englert, T. Györffy, and R. Werkmann, "Device-Bus u. Geräte-Interfacing", GSI, Darmstadt, Germany, 1997, unpublished.

[2] V. Rapp and W. Sliwinski, "Controls Middleware for FAIR", in *Proc. PCaPAC'14*, Karlsruhe, Germany, Oct. 2014, paper WCO102, pp. 4–6.

[3] K. Lüghausen, M. Dziewiecki, K. Kaiser, G. M. May, S. Rauch, and M. Thieme, "Next generation GSI/FAIR scalable control unit: lessons learned from 10 years in the field", in *Proc. PCaPAC2022*, Dolní Brežany, Czech Republic, Oct. 2022, pp. 76–78. doi:10.18429/JACoW-PCaPAC2022-THP15

[4] M. Thieme, W. Panschow, S. Rauch, and M. Zweig, "Single Board Computer for Device Control in the FAIR Accelerator Control System", presented at *ICALEPCS'09*, Kobe, Japan, Oct. 2009, paper TUP060.

[5] J. Serrano *et al.*, "The White Rabbit Project", in *Proc. IBIC'13*, Oxford, UK, Sep. 2013, paper THBL2, pp. 936–942.

[6] M. Kreider and T. Fleck, "FAIR Timing Master", in *Proc. PCaPAC'10*, Saskatoon, Canada, Oct. 2010, paper WEPL011, pp. 50–52.

[7] M. Arruat *et al.*, "Front-end software architecture", in *Proc. ICALEPCS'07*, Knoxville, TN, USA, Oct. 2007, paper WOPA04, pp. 310–312.

[8] A. Schwinn *et al.*, "FESA3 — The New Front-End Software Framework at CERN and the FAIR Facility", in *Proc. PCaPAC'10*, Saskatoon, Canada, Oct. 2010, paper WECOAA03, pp. 22–26.

[9] P. Gerhard, "About the New Linear Accelerator Control System at GSI", in *Proc. ICALEPCS'23*, Cape Town, South Africa, Oct. 2023, pp. 529–533. doi:10.18429/JACoW-ICALEPCS2023-TUPDP018

[10] K. Kaiser, "Description of the SIO SCU Slave Cards", GSI internal document nr. F-TN-C-xxx, Darmstadt, Germany, 2015.

[11] Modbus Organization, "Modbus Messaging on TCP/IP Implementation Guide v1.0b", 2006. https://www.modbus.org