THE GENERAL INTERLOCK SYSTEM (GIS) FOR FAIR

F. Ameil, C. Betz, GSI Helmholtz Centre for Heavy Ion Research, Darmstadt, Germany
G. Cuk, I. Verstovsek, Cosylab d.d., Ljubljana, Slovenia

Abstract

The Interlock System for FAIR, called the General Interlock System (GIS), is part of the Machine Protection System which protects the accelerator from damage by misled beams.

The GIS collects various hardware interlock signals from up to 60 distributed remote I/O stations through PROFINET to a central PLC CPU. Thus a bit-field is built and sent to the interlock processor (PC) via a simple Ethernet point-to-point connection. Additional software Interlock sources can be picked up by the Interlock Processor via UDP/IP protocol.

The Interlock System for FAIR was divided into two development phases [1]:

- Phase A: interlock signal gathering (HW and SW) and a status viewer.
- Phase B: fully functional interlock logic with support for dynamic configuration, interface with Timing System, interlock signal acknowledging, interlock signal masking, archiving and logging.

The realization of the phase A will be presented here.

INTRODUCTION

The Interlock System for FAIR, called the General Interlock System (GIS), is part of the Machine Protection System which protects the accelerator from damage by misled beams.

The GIS will be used as a slower part of the Machine Protection System for all machines comprising the FAIR accelerator complex. Three components of the Interlock System can be identified [2] (Figure 1):

- the operation layer,
- the processing layer, and
- the signal pickup layer.

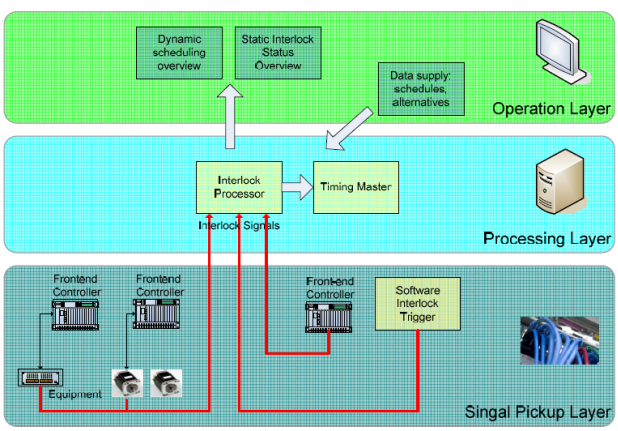

Figure 1: Layers of the interlock system.

The system must be able to collect hardware as well as software interlock signals from up-to 60 remote locations. Each remote station is capable of connecting to between 8 and 192 hardware interlock signals. Interlock transportation and processing must not take more than about 100 ms even when the system is extended to its limits - 4000 interlock signals. Based on these requirements, a combination of a PLC + interlock processor (PC) solution was chosen.

The purpose of Phase A was to confirm the selected architecture and components and to verify that such a solution is scalable and can reach the required performances. The solution will be installed and tested at the CRYRING accelerator in GSI.

The solution was developed by the company Cosylab d.d., Ljubljana, Slovenia, under supervision of the Control System Department at GSI.

SYSTEM DESCRIPTION

The main architecture of the GIS is presented in Figure 2 [3]. PLC Interface Modules (IM) with digital input modules are used to pick up hardware interlock signals. The control system network is used to pick up software interlock signals. After the GIS gathers both hardware and software interlock signals, it calculates summary interlock signals (specifics of this part will be defined and implemented in Phase B).

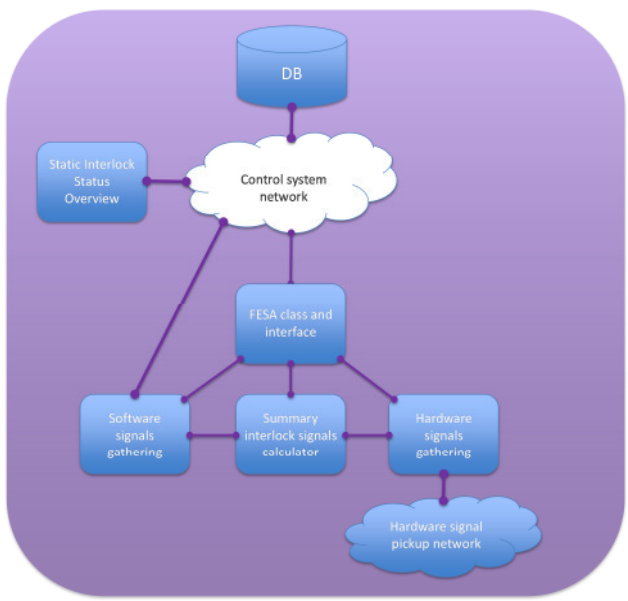

Figure 2: Architectural blocks of the GIS.

All changes of interlock signals are time-stamped. A user can use the GUI of the Static Interlock Status

ISBN 978-3-95450-148-9

Overview application to monitor currently active interlock signals. This application obtains information about interlock statuses by connecting to the FESA class interface of the GIS.

Hardware Signals Pick-up

In Phase A, four remote locations were implemented, each hosting an ET200 SP module with a different number of digital input modules for collecting hardware interlock signals (Figure 3). In total 192 hardware interlock signals can be picked-up. If the collection of interlock inputs in some areas is out of operation, these inputs are considered as if they are in an interlock state and the rest of the GIS system remains fully operational.

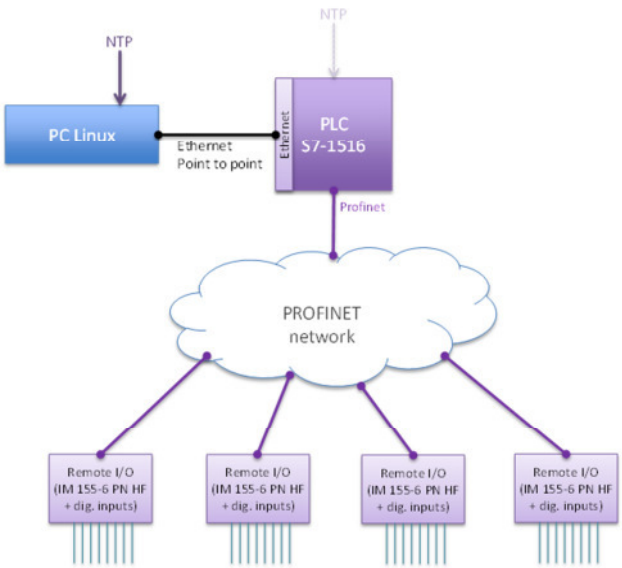

Figure 3: Hardware interlock signals pickup.

Remote stations are connected to a central CPU (S7 1516-3 PN/DP) via a Profinet network. The CPU runs with a fixed cycle time of 10 ms, thus defining the timestamp resolution. In each cycle, the PLC CPU gathers all hardware interlock signals and send them to Interlock Processor (PC) for further processing.

An extensive diagnostics is implemented to recognize different events and errors on the Profinet network and consequently put the relevant interlock signals into an interlock state. With those diagnostics, one can distinguish signalled interlocks from those caused by error, and furthermore the operator is able to see what kind of problem and in which location/module it occurred.

Software Signals Pick-up

Software Interlocks are collected by the Interlock Processor (PC) using UDP/IP as a communication protocol. For higher reliability, clients (systems sending software interlocks) send redundant datagrams whenever the interlock signal changes. The payload is encoded in such a way that critical bit-errors can be detected.

To make this interface and communication more reliable, the central station maintains a periodical communication (“heartbeat”) with all software interlock sources. An exception to this are interlock sources used for testing, for which the heartbeat functionality is not mandatory.

An interlock source can provide the status of its software interlock signal per Beam Production Chain in the same datagram. Also, if an interlock source provides many interlock signals, all of them can be signalled through a single datagram.

Interlock Status Overview Application

Phase A includes the Static Interlock Status Overview (SISO) application that serves for observing currently active interlock signals (Figure 4). The application connects to a GIS FESA class in order to obtain information about active interlocks, their type (hardware or software) and cause (signal or error) as well as their timestamps. Interlock signals are listed in the table and can be sorted or filtered by its name, type, cause or timestamp.

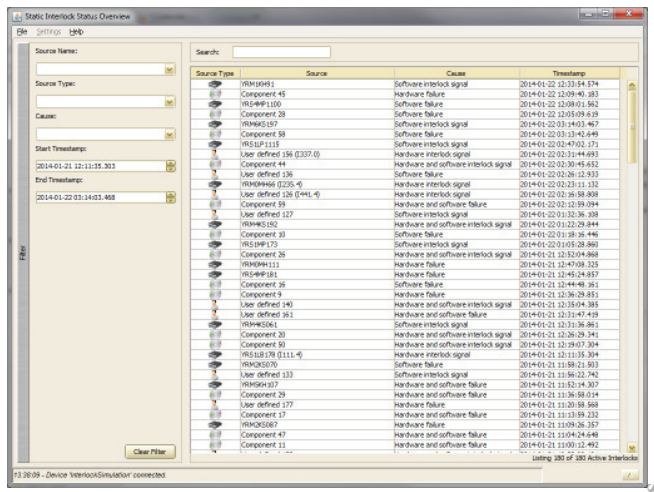

Figure 4: GUI of SISO application.

PHASE A - REALIZATION AND FAT

GIS phase A was developed by Cosylab. The system consists of a PLC CPU 1516-3 PN/DP and 4 remote stations based on IM 155-6 PN-HF, containing a total of 192 digital inputs (Figure 5). The PLC part of the system was developed in TIA portal v13. The rest of the functionality runs on the PC (not shown in the figure); it is implemented in C++ and integrated with the FESA class.

Implementation was followed by extensive factory acceptance testing with over 70 different unit tests covering all features of hardware and software interlock signals pickup, the FESA class and the GUI application.

ISBN 978-3-95450-148-9

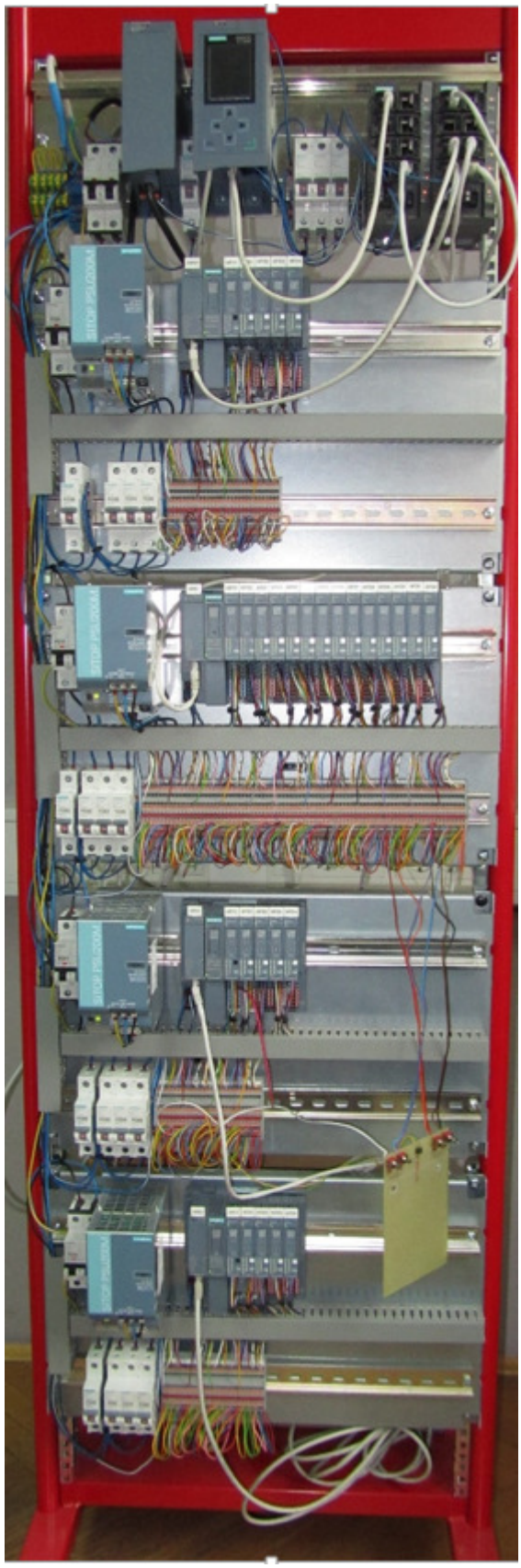

Figure 5: FAT configuration for GIS Phase A.

PERFORMANCE MEASUREMENTS

Two different measurements were done to confirm that the system complies with performance requirements:

- Cycle time measurement, using TIA portal.
- Response time, measured as difference between input and output signal change on two remote stations using an oscilloscope.

Phase A implements only a portion of the whole GIS system (4 out of 60 remote stations), so measuring performance of Phase A system only would not be enough. Therefore, a Profinet network simulator (SIMBA) was used to simulate the missing 56 remote stations with 3808 input signals (i.e. 4000 signals in total with the 4 physical remote stations from phase A).

With the cycle time measurement we want to show that system does not reach saturation even in the most intensive situations like failure of the remote station or even failure of the whole Profinet network. In normal operation, the GIS uses a fixed cycle time of 10 ms. In order to measure the effective cycle time, the fixed cycle time feature was disabled. Figure 6 shows the longest measured cycle time is 4 ms, therefore a fixed cycle of 10 ms has a safe margin.

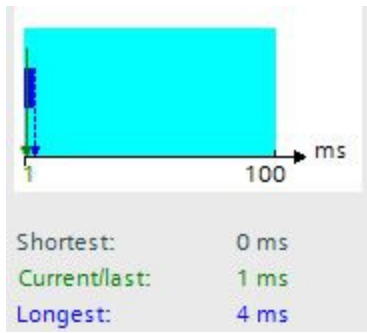

Figure 6: Cycle time measurement.

Real mitigation actions of the GIS will be implemented only in Phase B. To confirm that interlock transportation and processing takes less than 100 ms, we had to implement an additional feature of driving output signals on remote stations. This testing feature puts an output signal on the selected remote station(s) in a high state for a duration of one cycle each time any interlock signal change is detected. Measuring the time between the interlock signal going from high (OK) to low (NOK) state and the output signal going from low to high, gives the interlock system response time (which includes more than the initial requirement for transportation and processing time).

Different response times were measured because the result depends on when the change of input signal (done manually by opening contacts of the switch) occurs with respect to the start of the next PLC cycle [4]. Figure 7 shows the maximal measured response times (minimal response time was approx. 10 ms shorter). The pink line represents the interlock signal, transitioning from the OK to the NOK state. The yellow line is the response of the

ISBN 978-3-95450-148-9

GIS (response was simulated by driving an output signal on each remote station).

These results show that transportation and processing takes far less than 100 ms.

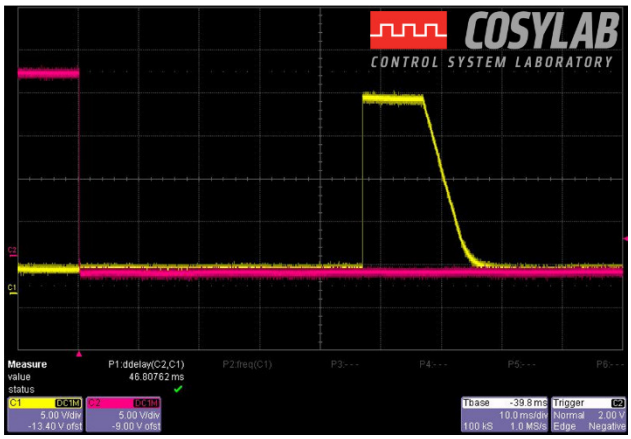

Figure 7: Response time measurement.

CONCLUSION AND OUTLOOK

The GIS for FAIR for Phase A was successfully developed and it fulfils the requirements. A Site Acceptance Test will be performed in the near future at CRYRING@GSI using the system developed for Phase A and operating under real conditions.

Another development is already in progress; its purpose is to include interlock state signals coming from other PLC systems (e.g. Vacuum Control System) using the I-Device functionality. Making use of this functionality avoids physically connecting many signals between two PLC systems; instead of physical signals, "logical" signals are exchanged via Profinet.

The system should be extended further in the next steps of the FAIR accelerator complex construction.

REFERENCES

[1] G. Čuk, FAIR Interlock System – Design Specification and Architecture, CSL-DOC-13-91257, version 2.1 (21.Nov.2013)

[2] J. Fitzek, G. Fröhlich, U. Krause: Detailed Specification of the FAIR Accelerator Control System Component "Interlock System", F-DS-C-08e, version 3.1 (11.Sep.2012)

[3] G. Čuk, FAIR Interlock System – Design document, CSL-DES-13-93467, version 1.4 (9.Apr.2014)

[4] G. Čuk, FAIR Interlock System – FAT Test Report, CSL-TRP-15-93918, version 1.0 (23.Apr.2014)

ISBN 978-3-95450-148-9