



**Detailed Specification of the
FAIR Accelerator Control System component
„FEC Device Classes“**

Document Name
F-DS-C-15e

Date yyyy-mm-dd
2011-08-10

Abstract

This document describes the Detailed Specification of the accelerator control system component "Equipment FESA Device Classes". This system is part of the "Control System Software Packages" work package and covers the PSP code 2.14.10.2.11.

Table of Contents

1.	Purpose and Classification of the Document	3
1.1.	Responsibilities	3
1.2.	Classifications of Requirements	3
2.	Scope of the Technical System	4
2.1.	System Overview	4
2.2.	Limits of the System and Environment	4
2.2.1.	Limits	4
2.2.2.	Interfaces	4
2.2.3.	Environment	4
2.3.	Basis of Concept	5
2.3.1.	Functional Requirements	5
2.3.2.	Non-functional Requirements	5
2.3.3.	General Constraints	5
2.3.4.	Architectural Principles	5
3.	Technical Specifications	6
3.1.	Basis	6
3.2.	FESA Device Classes	6
3.2.1.	Stepper Motor Control	6
3.2.2.	Serial Connection Equipment	7
3.2.3.	Proton-Linac RF-Systems	7
3.2.4.	Proton-Linac Specific Equipment for Ion Source Control	7
3.2.5.	Beam Position Monitor System, Closed Orbit Feedback System	8
3.2.6.	Pneumatic Drives	9
3.2.7.	Residual Gas Analyzers	9
3.2.8.	Summarized Vacuum System Representation	10
4.	Quality Assurance, Tests and Acceptance	11
4.1.	Development Methodology	11
4.2.	Quality Assurance System of the Supplier	11
4.3.	FAT	11
4.4.	SAT	11
5.	Documentation	12
6.	Warranty	12
7.	Scope of Delivery	12
I.	Attached Documents	13
II.	Related Documentation	13
III.	Document Information	13
III.1.	Document History	13

1. Purpose and Classification of the Document

The purpose of this document is to specify the Accelerator Control System work package "FEC Device Classes" (Front-End Controller Device Classes) for FAIR (PSP code 2.14.10.2.1).

The work package contains development of front-end software for connecting a list of equipment to the accelerator control system using the FESA (Front-end Software Architecture) framework. FESA is a comprehensive framework for the design, development, deployment and maintenance of front-end software. Originally created by CERN, FESA now is developed in a collaboration of CERN and GSI.

This document is the most detailed type of document in the hierarchy of Control System specifications.

Whenever regulations and requirements are specified in the General Specifications, Technical Guidelines or Common Specifications of the Control System they are only referenced in this document. The related documents are listed in Appendix II.

No legal or contractual conditions are treated in this document. All related information is given in the General Specifications for FAIR.

1.1. Responsibilities

The responsibilities with respect to changes and modifications of the present document are entirely in the hands of the Controls Department of the GSI Helmholtz Centre for Heavy Ion Research GmbH (GSI) Darmstadt.

For initial information please contact the administration of the Controls Department.

Further information on the organigram, names of responsible persons and task leaders, as well as the agreed document release and approval procedure is summarized in the organizational note 'Controls Project for FAIR'.

1.2. Classifications of Requirements

The following definitions of requirement classifications are being used throughout the document:

- **"Must"** or **"shall"** or **"is required to"** are used to indicate mandatory requirements, strictly to be followed in order to conform to the standard and from which no deviation is permitted.
- **"Must not"** or **"shall not"** mean that the definition is an absolute prohibition of the specification.
- **"Should"** or **"is recommended"** is used to indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others or that a certain course of action is preferred but not required.
- **"Should not"** or **"is not recommended"** mean that there may exist valid reasons in particular circumstances when the particular behavior is

acceptable or even useful, but the full implications should be understood and the case carefully weighted before implementing any behavior described with this label.

- **"May"**, which is equivalent to **"is permitted"**, are used to indicate a course of action permissible within the limits of the standard.

2. Scope of the Technical System

2.1. System Overview

On the lowest level of the control system, the FESA (Front-end Software Architecture) framework is used to connect any kind of equipment to the control system.

The FESA framework is used to generate so-called FESA classes, which contain the implementation for controlling a special kind of equipment (e.g. a FESA class for stepper motors). For each individual device, the FESA class has to be instantiated on the corresponding front-end controller.

Main task of the FESA device class besides the control of the underlying equipment is to represent a logical device (the set of properties, and their functionality, which the FESA class implements) that presents physical characteristics to the control system. Therefore, the FESA class contains interfacing to the equipment as well as functional behavior and the logical device interface representation towards the rest of the control system.

The work package contains a list of equipment, for which FESA classes are to be developed.

2.2. Limits of the System and Environment

2.2.1. Limits

The FESA class can only represent what is provided by the controlled equipment. Elaborate software constructs to overcome hardware limitations must be avoided.

2.2.2. Interfaces

Each FESA class has interfaces to its equipment, the client interface which is the logical device representation and interfaces to other components of the control system, like the interlock system, the alarm system, the post mortem system and the diagnostic logging system.

2.2.3. Environment

In general, the FESA class either runs on a dedicated front-end controller, or, in special cases, within the main computing center. For every FESA class, the exact environment has to be specified.

2.3. Basis of Concept

2.3.1. Functional Requirements

Specific functional requirements must be set up in accordance with the developer of the equipment and machine responsibilities, defining the respective type of usage of the equipment.

2.3.2. Non-functional Requirements

Specific non-functional requirements must be set up in accordance with the developer of the equipment and machine responsibilities, defining the respective type of usage of the equipment.

2.3.3. General Constraints

The system to be built must adhere to the guidelines and recommendations for software developments in the FAIR accelerator control system context, as referenced in the FAIR Common Specification F-CS-C-01e (Common Specification Accelerator Control System) [1]. The contractor has to adhere to the FAIR Technical Guidelines [2][3][4][5], to the Equipment Integration Guideline [8] in general and the FAIR development guideline for FESA classes [6] in particular.

The supplier of the work package must identify the relevant standards and recommendations before start of the development. Details must be fixed as part of the technical design concept in the initialization phase.

The FESA classes must be based, as far as possible, on the tools and mechanisms which are provided by FESA.

Each FESA device class must be compiled with the FESA front-end framework as provided and supported by the control system main contractor (see [1]).

2.3.4. Architectural Principles

The respective classes must implement an abstraction layer which represents the equipment in physics oriented way, and must not be limited to direct pass-through of the equipment's parameters.

Besides converting technical units of the equipment (like ADC data) into physical units (like positions, given in mm, or currents, given in ampere), the FESA class should combine, and rearrange, data and activities. E.g. a FESA class handling two motors of a slit should allow moving both motors simultaneously with one command. Movement should be parallel (shifting the slit) or in opposite direction (changing the slit width).

The Software Architecture Guideline for the Control System [7] fully applies.

3. Technical Specifications

3.1. Basis

For each FESA class, specific requirements exist and cannot be fully listed here, due to the fact, that the equipment that needs to be controlled is not yet present.

As soon as the details of the equipment to be controlled are fixed, a detailed technical design must be created. This technical design must consider the interaction with the equipment, the real-time behaviour, and the logical representation by a set of properties. The break down structure, which means handling the functionality in single or multiple FESA classes, must be fixed. The technical design must include the integration of the FESA class into the control system, i.e. the connection of the FESA class to central systems like the Interlock System, the Alarm System, the Post Mortem System, and the Diagnostic Logging System.

Before start of development, the technical design must be approved by the FAIR contracting body, represented by the control system main contractor (see [1]).

3.2. FESA Device Classes

The work package contains a set of FESA classes which either require special knowledge of the underlying front-end controller system, or which handle unique pieces of equipment, of which only one or a few exist in the FAIR facility.

The work package contains development of FESA classes and the appropriate specific equipment drivers for the subsystems listed below.

3.2.1. Stepper Motor Control

In the FAIR facility in many locations stepper motor systems are used to precisely adjust movable elements. Stepper motors are mostly used to precisely adjust beam diagnostic insertion elements into the beam.

Industrial type control FECs will be used to handle stepper motors in the accelerator control system. These FECs provide an internal controller which will run the FESA stepper motor control classes. Each one of these stepper motor control FECs can handle up to 8 motors.

The stepper motor front-end software has to implement a unified handling and representation of stepper motors. It must handle single motors, e.g. for scrapers, and it must handle simultaneous movement of several motors, e.g. when they are combined to drive slits or collimators. In any case, the FESA class has to interact closely with the stepper motor controller electronics, implemented in the stepper motor control units.

3.2.2. Serial Connection Equipment

Commercial off the shelf equipment often provides serial interfaces like RS-232, RS-422, RS-485 or IEEE-488.2 GPIB. To connect such equipment to the accelerator control system, without the need for modification of the equipment, an industrial type controller FEC is used. Providing several serial connections, each FEC can handle several pieces of equipment by different connections.

For each type of equipment which is interfaced by a serial connection to the industrial type front-end controller one or, in special cases, several FESA device classes have to be developed. The device class depends on the type of equipment to be controlled and on the usage of the equipment in the facility. Besides such specifics, common access mechanisms for communication by a serial connection must be developed which will then be used in all FESA device classes for serial connected equipment.

3.2.3. Proton-Linac RF-Systems

Reliable operation of the proton linac RF systems requires several low level control loops to stabilize the RF output. For this low level RF control a specific controller is envisaged (Libera-RF controller) which provides a powerful internal processor. This internal processor can in addition act as accelerator control system front-end controller, hosting the FESA equipment control class.

Appropriate FESA front-end software to interface the proton linac RF systems has to be developed. This front-end software has to be finely tuned and carefully adjusted to the specifics of the local control RF control system, implemented on the RF controller.

3.2.4. Proton-Linac Specific Equipment for Ion Source Control

Operation of the ion source of the linac requires handling of different equipment like voltage and current power supplies, servo motors for gas inlet and a microwave RF system. Many of these pieces of equipment are specific because they are unique in the facility.

Although most of the equipment implements only few set points, handling is complicated because all parts of the ion source must operate closely together. The equipment therefore has to be interfaced by a harmonized set of FESA classes. Development of the FESA front-end software for the ion source components therefore requires a good overall view of the interaction of all ion source components.

3.2.5. Beam Position Monitor System, Closed Orbit Feedback System

The beam position monitor (BPM) system allows the non-intercepting determination of the beam position (centre of mass). It consists of a number of BPM pickups and corresponding processing units. In order to integrate the BPM system into the control system, corresponding FESA classes are required which support parameterization of the BPM system, read-out of processed measurement values, and triggering of calibrations with several calibration modes.

Since BPM systems with different characteristics exist for rings and straight sections (beam transport sections and linacs), different FESA classes must be provided that handle the special requirements of the different systems. It is recommended to investigate before the realization, if a common base FESA class for all BPM systems containing the common parts can be used.

In addition to the FESA classes for single BPMs, FESA classes for potentially needed data concentrator components are also part of the delivery.

BPM system for ring machines

Within ring accelerators, the BPM system is in general responsible for the determination of the transversal position of the bunched ion beam, for the measurement of closed orbit on a bunch-by-bunch manner and optionally for the tune measurement. The BPM system itself is specified in [9] (for SIS100) and in [11] (for CR). It is foreseen that the corresponding FESA class will be running on the appropriate processing units of the BPM system if possible. An existing prototype implementation was set-up at GSI in the last years including FESA classes and can be used as basis for concept. Additional requirements for FAIR like correct handling of bunch manipulations must be supported.

The FESA class (or classes) must provide access to single BPMs and to an abstraction of the whole BPM system for a given ring (recommended to be represented as separate device). The parameterization of the BPM system must include the possibility to set information about e.g. the beam production chains, beam characteristics, bunch manipulations (merging, rebunching) originating from the settings management component of the accelerator control system.

BPM system for straight sections

In straight sections (HEBT, Linac, PBar Target, SFRS), the BPM system is used in a so-called single path mode to detect beam position and longitudinal bunch shape. It includes means to notice possible drifts of the magnetic field of the beam transport system by storing and comparing measured beam positions. The BPM system for straight sections is specified in [10] (for HEBT, pbar Target and SFRS) and in [12] (for p-Linac).

Again, the FESA class (or classes) must provide access to single BPMs and to an abstraction of the whole BPM system for a given section (recommended to be represented as separate device). The FESA class must allow parameterizing the BPM systems with all needed information originating from the settings management component of the accelerator control system.

Closed orbit feedback system

The goal of the closed orbit feedback system for the SIS100 is to achieve a higher acceptance of the accelerated beam and to prevent resonance effects. The closed orbit feedback system itself is specified in [13]. It calculates corrections for the corrector magnets taking into account information about the different beam production chains and the desired position of the beam as well as the current beam position measured by the BPM system of the ring.

The feedback system is a dedicated hardware system, which must be controlled from a corresponding FESA class. Besides other functionality, the FESA class must provide means to start/stop the correction process, to read out the applied correction values, to log the correction values and to provide those values to the post mortem system. The FESA class must also support setting the necessary set values, e.g. beam characteristics and desired beam position.

3.2.6. Pneumatic Drives

Pneumatic drives are used all over the FAIR facility. E.g. for beam diagnostic devices, the corresponding pneumatic drives move the diagnostic instruments into their measurement position in the path of the ion beam or vice versa back in a save park position where they do not disturb the beam. Pneumatic drives are specified in [14].

In order to set and read positions of pneumatic drives, an appropriate FESA class must be provided. The FESA class must integrate the pneumatic drives into the control system, providing means for driving and reading out position and state as well as realizing the connection e.g. to the interlock, alarm and logging system. On a technical level, the FESA class must realize the interface to the pneumatic drive, e.g. the direct control of the PLC using the IEPLC interface.

It is recommended to provide one generic FESA class that can be used for all pneumatic drives, however, special cases like associated pneumatic drives which must be driven in an coordinated way shall be supported as well.

3.2.7. Residual Gas Analyzers

The need for extremely good vacuum conditions in the accelerators requires detailed analysis of the components of the residual gas in the facility. Commercial residual gas analyzers will be used to determine the composition, and the partial pressure, of the remaining particles in the vacuum tubes. To offer the results of the residual gas analysis or broad usage in the facility, the analyzers have to be interfaced to the accelerator control system.

Such residual gas analyzers offer a lot of functionality. Rather than representing the knobs and meters of the equipment one by one in the accelerator control system, an abstraction is required. Reducing the full functionality to the very needs for the operation of the facility allows interpretation of the analysis also by non-experts of the vacuum system, like operators in the main control room.

The abstraction must be implemented in the front-end control FESA class. A FESA class must be developed, which offers the functionality which is essential

for the operation of the facility in a comprehensible manner which is understandable for the machine operation staff.

3.2.8. Summarized Vacuum System Representation

The vacuum system operation of the FAIR facility will be based on a SCADA system. At first step, the front-end control and the industrial control are distinct domains in the accelerator control system. However, the need for a homogeneous view of the complete facility requires connections between these two domains. The accelerator control system will support representing equipment which is handled by the industrial control domain by FESA device classes like any other equipment interfaced by the front-end control part of the accelerator control system.

Main parameters of the vacuum control system, which are of interest for the machine operation crew, must be represented by appropriate FESA devices. The topic "summarized vacuum system representation" subsumes development of FESA classes for representation of the main parameters of the vacuum system.

4. Quality Assurance, Tests and Acceptance

4.1. Development Methodology

Main task before starting to develop is to decide together about the interface towards the control system, about the logical device representation. The FESA class design including this interface has to pass a design review, as described in the guideline for connecting new equipment to the control system [8]. The FESA classes must be developed in adherence to the guideline for designing FESA equipment software at the GSI and the FAIR facility [6]. All aspects of these guidelines apply.

Each FESA class must be developed iteratively. At start of development, for each type of equipment the technical design concept and the iteration plan must be established, and must be approved by the FAIR contracting body. The iteration plan must also fix content and duration of the first iteration step.

Each iteration cycle must result in a running system which can be evaluated, and tested, at FAIR site. The first iteration, which has to be available as early as possible, must concentrate on the most critical functionality. In successive iterations, the system is enhanced by adding features until the desired total functionality is reached.

At end of each iteration cycle the achieved status of the system will be evaluated and the iteration plan will be adjusted. Each iteration cycle must be approved by the FAIR contracting body before it can be started.

4.2. Quality Assurance System of the Supplier

The Common Specification "Accelerator Control System" [1] fully applies.

4.3. FAT

The Common Specification "Accelerator Control System" [1] fully applies.

For all FESA classes, it has to be decided and agreed upon in advance, if the FAT can be done at the contractors site, or if the FAT needs to be done at the contracting body's site (due to equipment, that is already installed within the FAIR accelerator complex). If the FAT needs to be done at the contracting body's site, the FAT has to be announced well in advance to be able to access the corresponding equipment if necessary.

4.4. SAT

The Common Specification "Accelerator Control System" [1] fully applies.

Main part of the site integration test is the test of the FESA class together with the real equipment.

Part of the SAT is a test of the FESA class functionality itself. For ongoing integration tests, even when equipment is not working properly, it is necessary to have mock devices of the corresponding FESA class that behave, as if the real equipment were present (e.g. values can be set and read, the values are kept in memory and not sent to the equipment).

5. Documentation

The Common Specification "Accelerator Control System" [1] fully applies.

6. Warranty

The conditions and warranty period specified in the contract applies.

7. Scope of Delivery

The scope of delivery of this work package comprises

- FESA classes
- Unit tests for developed classes
- Utilities for handling and testing of the FESA class, if appropriate
- Source code and build instructions of delivered software
- Documentation of the delivered software, including description of architecture and outline of the classes

GUIs for usage in normal operation (GUIs running in the main control room) are not part of the delivery.

I. Attached Documents

List of abbreviations for controls (Abbreviations_Controls.pdf).

II. Related Documentation

- [1] F-CS-C-01e, FAIR Common Specification "Accelerator Control System"
- [2] F-TG-C-03e, FAIR Technical Guideline "Control System Equipment Interlock and Status Signal Interface"
- [3] F-TG-C-01e, FAIR Technical Guideline "Ethernet Network Connectivity"
- [4] F-TG-C-02e, FAIR Technical Guideline "Control System Equipment Control Interfaces"
- [5] F-TG-C-04e, FAIR Technical Guideline "Control System Equipment Functional Requirements"
- [6] F-DG-C-01e, "FESA Development Guideline"
- [7] F-DG-C-03e "Software Architecture Guideline"
- [8] F-DG-C-04e "Equipment Integration Guideline"
- [9] F-DS-BD-22e, FAIR Detailed Specification "POS+BPM_1S"
- [10] F-DS-BD-23e, FAIR Detailed Specification "POS+BPM_T+PT+F"
- [11] F-DS-BD-24e, FAIR Detailed Specification "POS+BPM_CR"
- [12] F-DS-BD-25e, FAIR Detailed Specification "POS+PHP+TOF+BPM_PL"
- [13] F-DS-BD-29e, FAIR Detailed Specification
"Special+ClosedOrbitFeedback_1S"
- [14] F-DS-BD-47e, FAIR Detailed Specification "SubSys+PneumaticDrive"

III. Document Information

III.1. Document History

Version	Date	Description	Author	Review / Approval
0.1	19. Sep. 2011	Draft version	U. Krause	
0.1	06. Oct. 2011	Draft version	U. Krause	
1.0	07. Oct. 2011	Final draft	CCT	CCT
1.1	17. Oct. 2011	Adapted to new structure	U. Krause	
2.0	18. Oct. 2011	Final version	CCT	CCT
2.1	17. Nov. 2011	Renaming of referenced guidelines	CCT	
3.0	03. Aug. 2012	Incorporated FAIR review comments	CCT	