

# Architectural concept for the ITER Plasma Control System

W. Treutterer<sup>1</sup>, D. Humphreys<sup>2</sup>, G. Raupp<sup>1</sup>, E. Schuster<sup>3</sup>, J. Snipes<sup>4</sup>, G. De Tommasi<sup>5</sup>,  
M. Walker<sup>2</sup>, A. Winter<sup>4</sup>

<sup>1</sup>Max-Planck Institute for Plasma Physics, EURATOM Association, Garching, Germany,  
Wolfgang.Treutterer@ipp.mpg.de, Gerhard.Raupp@ipp.mpg.de

<sup>2</sup>General Atomics, San Diego, CA, United States of America,  
humphreys@fusion.gat.com, walker@fusion.gat.com

<sup>3</sup>Lehigh University, Bethlehem, PA, United States of America,  
schuster@lehigh.edu

<sup>4</sup>ITER Organization, 13115 St. Paul-lez-Durance, France,  
Joseph.Snipes@iter.org, Axel.Winter@iter.org

<sup>5</sup>CREATE/Università di Napoli Federico II, Napoli, Italy  
detommas@unina.it

The plasma control system is a key instrument for successfully investigating the physics of burning plasma at ITER. It has the task to execute an experimental plan, known as pulse schedule, in the presence of complex relationships between plasma parameters like temperature, pressure, confinement and shape. The biggest challenge in the design of the control system is to find an adequate breakdown of this task in a hierarchy of feedback control functions. But it is also important to foresee structures that allow handling unplanned exceptional situations to protect the machine. Also the management of the limited number of actuator systems for multiple targets is an aspect with a strong impact on system architecture. Finally, the control system must be flexible and reconfigurable to cover the manifold facets of plasma behaviour and investigation goals.

In order to prepare the development of a control system for ITER plasma operation, a conceptual design has been proposed by a group of worldwide experts and reviewed by an ITER panel in 2012. In this paper we describe the fundamental principles of the proposed control system architecture and how they were derived from a systematic collection and analysis of use cases and requirements. The experience and best practices from many fusion devices and research laboratories, augmented by the envisaged ITER specific tasks, build the foundation of this collection. In the next step control functions were distilled from this input. An analysis of the relationships between the functions allowed sequential and parallel structures, alternate branches and conflicting requirements to be identified. Finally, a concept of selectable control layers consisting of nested “compact controllers” was synthesized. Each control layer represents a cascaded scheme from high-level to elementary controllers and implements a control hierarchy. The compact controllers are used to resolve conflicts when several control functions would use the same command signals as their outputs. They consist of a collection of potentially conflicting control functions from which one at a time is exclusively activated by a mode selector signal.

It can be shown that this architectural design is capable of implementing all of the presently known functional control requirements. Furthermore, this design takes already into account that the result of future experiments at ITER will create additional requirements on the functions or performance of ITER plasma control.

Keywords: real-time control, event detection, exception handling, protection, system architecture

## 1. Introduction

The system architecture of the ITER Plasma Control System (PCS) comprises several aspects. The functional architecture describes the system from the perspective of control functionalities and their interdependences. The communication concept focuses on the interaction between function components. The software architecture, finally, provides the fundament on which the implementation of control functionality and communication rely. The scope of this contribution is confined to the functional architecture.

To develop an adequate design for the ITER PCS a standard method of system engineering has been adopted [1]. As a preparation for the architectural design fusion and control system experts of worldwide fusion laboratories have gathered principal high-level functional requirements [2, 3, 4, 5, 6, 7]. Physics-based control functions were in the primary focus, but planned and expected methods of operation have also been considered. This article essentially summarises the results leading to the functional PCS specifications related to system architecture in [7].

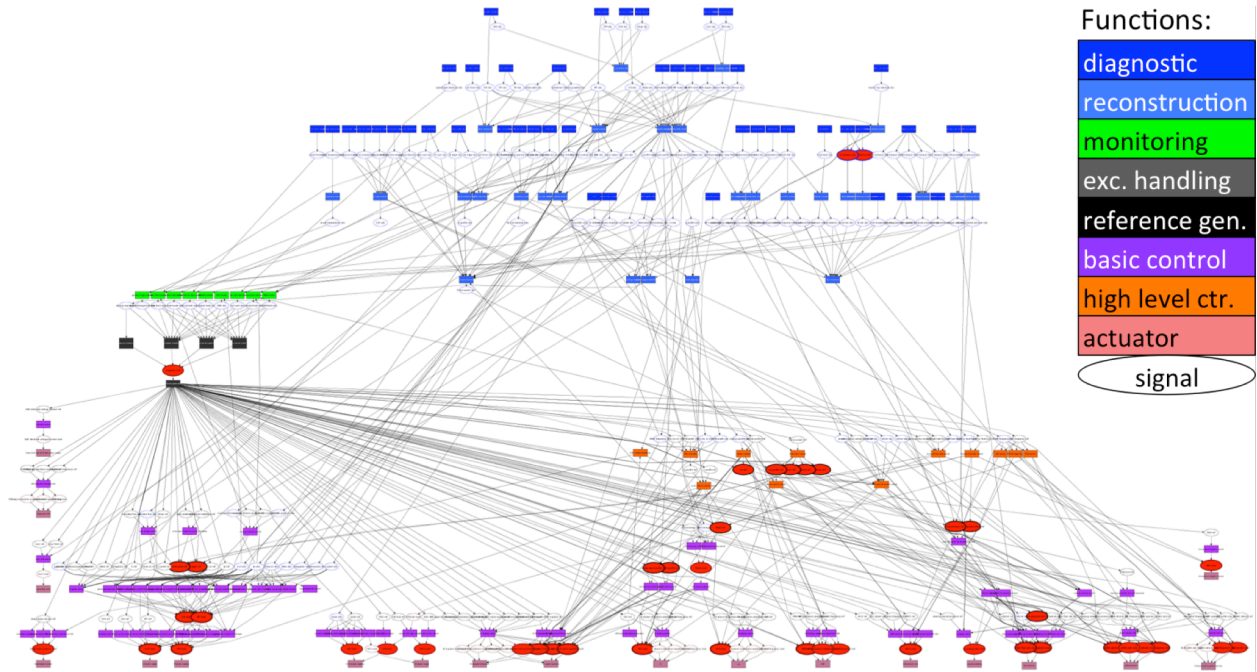


Fig. 2: Dependency graph of ITER PCS control functions, conflicting requirements are spotted by red ovals

Section 2 shows, how the control functions were analysed. Dependencies were unfolded and potential conflicts identified. Furthermore, functions with common scope and comparable features could be assigned to basic domains like diagnostics, high-level and elementary control, actuator management, exception handling and reference generation. The analysis, however, also revealed, that a fixed structure of functional blocks does not fit for all functions. Almost all high-level control functions follow independent targets but make use of the same limited set of elementary controllers and actuators. A possible resolution is presented in Section 3. With the introduction of two concepts structure can be organised variably. Compact controllers and control layers permit to switch control functions and combinations of these. Together with the operation requirements a strawman architecture design has been proposed which is detailed in Section 4. At least two already existing control system frameworks, MARTE (JET) [8] and DCS (ASDEX Upgrade) [9] seem to be capable to realise the architectural concept for ITER PCS.

## 2. Function analysis

### 2.1 Control function collection

In a joint attempt fusion experiments and research laboratories assembled their expertise devising use cases and required physics control functionality for plasma pulses at ITER. The incoming records were first subject to a consolidation procedure. Duplicates were removed. Necessary but not explicitly listed functions like measurement and reconstruction of physics quantities, used by a control function, were amended. Finally, the condensed list of control functions amounted to 190 entries. Note, that at this conceptual stage of design only very abstract, high-level function descriptions are of interest. The actual number of control functions in ITER

PCS will be much larger, once these descriptions will be broken down to finer granularity at detail level.

All control functions associate a certain context describing the input quantities used and the resulting output, as well as conditions under which they may or should be applied. With exception of components connected to external diagnostic and actuator systems, all control functions use results from other functions and provide input to subsequent components, thus forming threads processing measurement data to actuator commands.

Current plasma control systems already deal with a number of such threads, which are independent enough to be executed in parallel. Parallel execution, in turn, allows for reducing the overall response latency and scaling up system to higher functionality and performance. While this is also true for the ITER PCS, advanced automation of control will interweave the control functions to a much larger degree than today.

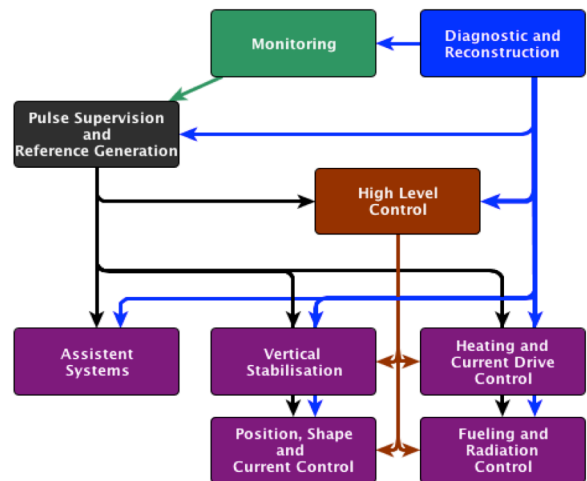


Fig. 1: Simplified dependency graph with function domains

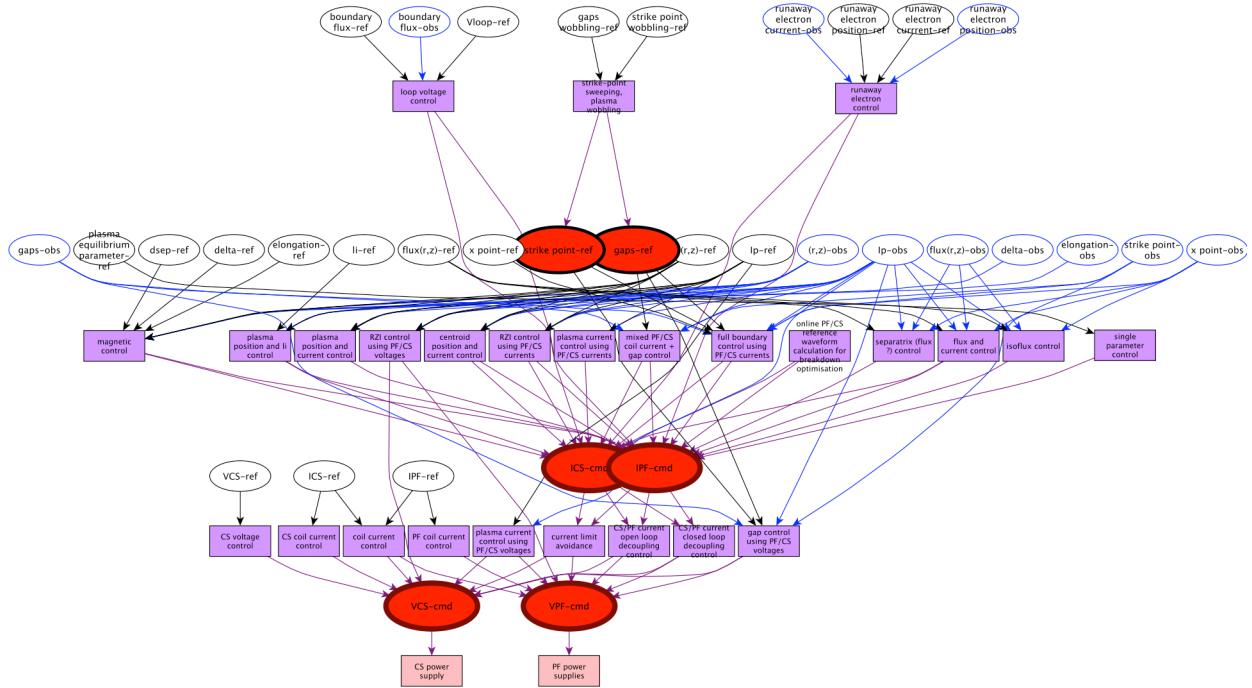


Fig. 3: Conflicts in plasma position, shape and current control

Nowadays, experienced pulse designers manually translate tuning recipes for plasma optimisation to low-level command waveforms in the pulse schedule. In ITER this task will be adopted by high-level functions, such as burn control and heat load control.

## 2.2 Dependencies

The organisation of threads is a major factor for the architectural design. Therefore, the dependency structure of control functions has been analysed in more detail. Based on the function descriptions input and output data were modelled as signal entities yielding over 270 signals linking the 190 control functions. Fig. 1 visualises the dependency structure in a graph. For better understanding, the functions have been drawn as colour coded boxes and clustered to a few basic domains, namely diagnostic and reconstruction (blue), feedback control (brown and magenta), actuation (light red) and monitoring (green). Reference generation and exception handling are shown in black. The white and red ovals stand for signals. The red colour marks cases where a quantity can result from different control functions. Such conflicts designate functional alternatives and are important architectural drivers (see Section 3.2).

Fig. 1 with its strongly interweaved links between control functions highlights the challenges to the architectural design. Some fundamental findings can already be derived:

- With very few exceptions exclusive assignments of diagnostic and reconstruction functions to dedicated feedback control functions are not possible. This suggests keeping those groups separate.

- Feedback control is organised in hierarchies such that the architecture has to take nested control loops into account.
- Monitoring functions (green) and reference generation (black) play central roles in that they are gathering and distributing information throughout the entire system. This graphical dominance is the counterpart to the importance of their coordination function for the overall system.

## 3. Architectural design

The challenges revealed in the previous analysis are main drivers in the architectural design. In order to structure the system and to enable the definition of basic interfaces for a later conversion to software modules, it is necessary to define domains with related and associated functions. This step requires, however, that the dependency conflicts get resolved.

### 3.1 Functional domains

The simplified diagram in Fig. 2 gives a clearer picture identifying the major function groups for architectural design. Event detection and exception handling will play a major role in plasma control for ITER. Since the reference provides guidance to the control algorithm, the functional responsibility to define reaction policies on exceptions has been assigned to the reference generation function, although the implementation of the policies can be distributed and embedded in any of the control system components. Corresponding to the different roles, the functional architecture outlined here distinguishes two major topics: Pulse Supervision and Pulse Control.

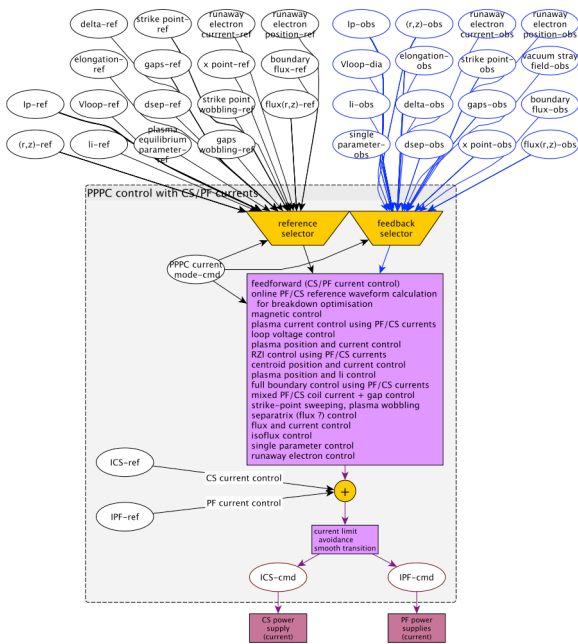


Fig. 4: Compact controller for position current and shape

Pulse Supervision represents reference generation extended with central exception handling. Reference signals in general have the function of coordinating control functions even if these functions do not have explicit linkage. Under normal circumstances, reference values for the actual point in time can be generated by interpolation of waveforms laid down in the pulse schedule. Beyond that, exception handling is given the superior authority to modify the references according to pre-defined rules. That way, it combines the individual reactions of control functions into a consistent response. Further exception handling in the form of localised repair and degradation actions can be distributed in all of the Pulse Control components.

Pulse Control represents the other components of the control loop, with exception of the controlled system, which is external to PCS. It can be further decomposed in a diagnostic section including measurement conditioning, physics quantity reconstruction, monitoring and forecasting, in a feedback control and in an actuator management section. Out of those, the feedback control section is the most demanding one in terms of architectural design.

### 3.2 Conflict resolution

A fusion device like ITER requires the control of an abundance of variables, which are cross-coupled to a considerable degree. The analysis of the control function dependencies in Section 2.2 revealed conflicts in the output signals shown as red ovals in Fig. 1. In case of reconstruction functions, these conflicts result from redundant information gathered by different sensing methods. They are generally not critical, instead they can be combined e.g. by integrated data analysis and Bayesian filters to calculate quantities with improved quality. Most conflicts are found in the feedback control

domains (in orange and magenta colours) and there, in particular, at the actuator commands. This observation corresponds also to the issue of actuator management. Conflicts in feedback control functions mostly result from divergent control goals or from control scheme variants, which are mutually exclusive. The system architecture must provide structures and methods to resolve such conflicts in order to allow a predictable and reproducible behaviour of PCS.

#### 3.2.1 Compact controllers

A possible resolution strategy can be illustrated on the example of the Plasma Position, Shape and Current Controller domain. Fig 3 shows the initial situation, where conflicts appear for reference signals (“\*-ref”) for controller outputs (“\*-cmd”). The latter often also serve as commands to actuators.

A closer look at the respective control functions shows, that they mostly differ in the controlled variables but use the same actuators. In the case of the ITER magnetic coil system only two solutions seem reasonable:

- If conflicting control functions correspond to control goals for different scenarios or plant and plasma states, sequencing in time can separate them.
- If conflicting control functions represent concurrent control goals they must be combined by designing a corresponding multivariable controller.

A monolithic multi-input/multi-output (MIMO) controller seems to be unmanageable, due to the variability and nonlinearity of plasma physics and actuators. Instead, collapsing all control functions connected to the power supplies into one “compact controller” offers a reasonable solution. A control mode defined as a reference signal from Pulse Supervision is used to select the respective active control algorithm together with the associated reference and feedback signals (Fig. 4). Algorithms may be linear as well as non-linear, single input-single output as well as multivariable and simple as well as complex. In addition, a compact controller incorporates a feed forward path as well as output conditioning with saturation and load levelling, as these features need tight interaction with the control algorithms.

The same strategy can be applied to the other basic control systems. Examples are inductive and non-inductive plasma current control, vertical stabilisation, position and shape control, density control, impurity control, bulk heating control and temperature profile control. In summary, they form a class of elementary controllers taking care of simple subsystems. Their functions are almost orthogonal and have little cross coupling to other systems. Used in that way, elementary controllers contribute to simplify the overall system model by implicit linearisation as well as by compensation of model uncertainties, cross-coupling and external disturbances. Such elementary controllers exist in almost every conventional fusion experiment.

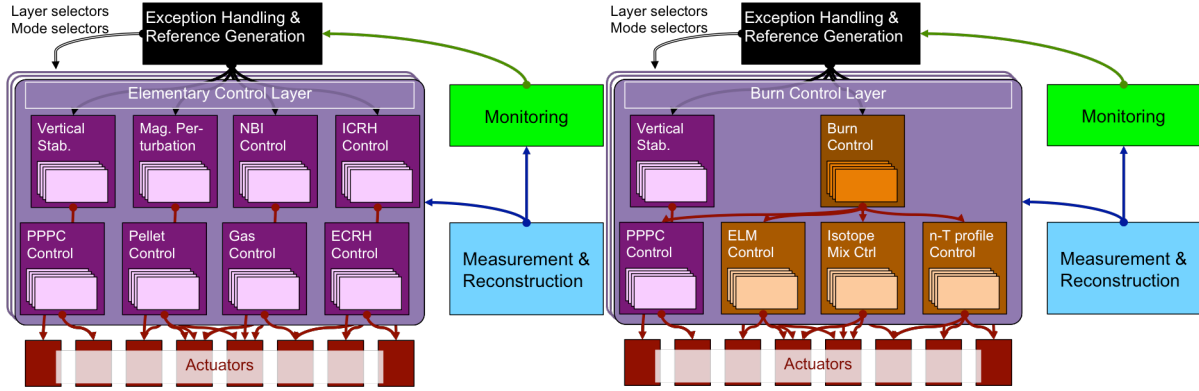


Fig. 5: Control layers with focus on different level of control automation

### 3.2.2 Control hierarchies

An outstanding control domain in the function dependency graph is high-level control. It contains functions like NTM control, burn control or ICRH coupling feedback whose primary goal is to prevent disruption counteracting plasma instabilities and optimise performance. This task requires exploiting also the full knowledge of correlation between plasma quantities, their spatial distribution and their dynamic evolution. The target can be achieved only with an integrated view on the controlled system. Characteristically high level control functions compute modified references for plasma shape and kinetic profile controllers, often across the boundaries of the previously identified control domains.

With compact controllers it is also possible to form such nested control loops. A control hierarchy can be defined as a control mode, which uses the output of a superordinate controller as reference signal.

### 3.2.3 Control layers

A couple of higher-level kinetic control functions, namely temperature or pressure profile control makes use of combinations of the heating systems. Here, an integrated compact controller tying all heating facilities is more suitable than an agglomeration of elementary controllers for each single heating system. It can deal better with the limitations in the heating and current drive capabilities at pulse execution time such as automatic power source replacement or load distribution.

In between both extremes a number of variations consisting of combinations of elementary and integrated controllers exist. Architectural design should allow any of such choices.

A possible method to achieve this can be the introduction of the novel concept of “control layers”. A control layer would comprise a pre-defined combination of elementary and integrated controllers and could be activated with a mode selector analogously to the compact controllers (Fig. 5).

Depending on the goal of a pulse phase, control layer contents can be defined such that the actual control

function requirements are mapped on a temporary controller structure comprising an adequate selection of compact controllers and respective control modes. While compact controllers resolve conflicts between individual control functions, control layers resolve conflicts between control structures.

## 4. Strawman architecture

The set of control architecture requirements paired with the mandatory ability to adapt to new future functional and technological development can best be satisfied with a modern framework architecture consisting of re-usable modular components. Such frameworks also exist in the domain of fusion plasma control. In particular, the ASDEX Upgrade Discharge Control System (DCS) [9] and the MARTE real-time control framework [8] developed at JET and used by a number of other facilities have properties enabling them to implement the fundamental architectural functions. Both systems share the same basic idea: functional components form strictly separated modules only connected via their input and output signals. Signal distribution, in turn, is managed by a separate component. This allows establishing connections easily between control functions just by configuration data without changing the function algorithm. DCS even comes with an implementation of customisable compact controllers [10]. Control layers are not yet found in the two frameworks but could be implemented.

## 5. Conclusion

Analysis of the needed functionality to control an ITER plasma pulse revealed a complexity considerably exceeding that of present experimental fusion devices. In particular, nested control hierarchies, variable control structures and exception handling are required for highly automated operation. The combination of compact controllers, control hierarchies and control layers forms an architectural toolset covering already all identified feedback control functions. Even more important, it has the capability to also integrate the yet unknown future complex control functions.

The functional architecture can be implemented based on the already existing modern modular control frameworks. These are also capable of satisfying the non-functional requirements that have not been detailed

in this contribution. In summary, it can be expected with reasonable confidence that a system architecture satisfying the presently assumed functionality can be developed timely for the preparation and operation of ITER.

## References

- [1] T. Weillkiens, "Systems Engineering with SysML/UML", Morgan Kaufmann, 2008, ISBN 978-0-12-374274-2
- [2] J. Snipes, "Physics of the Conceptual Design of the ITER Plasma Control System" (in this issue)
- [3] A. Winter, "Design Progress of the ITER Plasma Control System" (in this issue)
- [4] J. Snipes, "PCS Design Description Document", v 1.4, ITER IDM UID BRL9JX
- [5] J. Snipes, "Plasma Control System Physics Requirements Document", v. 1.5, ITER IDM UID BH2429
- [6] A. Winter, "PCS Requirements Specification", v. 1.4, ITER IDM UID BX6BL3
- [7] A. Winter, "PCS Functional Specification", v. 1.6, ITER IDM UID BRG63G
- [8] Neto, A.C.; Sartori, F.; Piccolo, F.; Vitelli, R.; De Tommasi, G.; Zabeo, L.; Barbalace, A.; Fernandes, H.; Valcarcel, D.F.; Batista, A.J.N., "MARTE: A multiplatform real-time framework", IEEE Trans. Nucl. Sci. 57 (April(2)), 2010, 479-486, DOI: 10.1109/TNS.2009.2037815, <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5446608&isnumber=5446471>
- [9] W. Treutterer, G. Neu, G. Raupp, D. Zasche, T. Zehetbauer, ASDEX Upgrade Team, "ASDEX Upgrade's New Plasma Control Scheme", Fusion Eng. Des. 81, 15- 17 (2006), 1927-1931, doi:10.1016/j.fusengdes.2006.04.010
- [10] W. Treutterer, T. Zehetbauer, V. Mertens, G. Neu, G. Raupp, D. Zasche, ASDEX Upgrade Team, "Plasma Feedback Controller Reorganisation for ASDEX Upgrade's New Discharge Control and Data Acquisition System", Fusion Eng. Des. 74 (2005), 501-505, DOI: 10.1016/j.fusengdes.2005.06.151