Abstract

An emulation system is described which replaces the ASDEX-Upgrade tokamak device for controlled nuclear fusion as a data source and its discharge control system as a processing unit. The purpose of the system is to generate real time data which are either preprogrammed or recorded from previously executed plasma discharges, and to process them like the discharge control system would do. The real time emulation of both, fusion device and control system, allows enhanced flexibility in testing and analysing the software of the control loop. This development follows the trend towards safe control concepts for future reactor control systems.

I. INTRODUCTION

ASDEX-Upgrade is an experimental plasma physical device of the tokamak type for controlled nuclear fusion. Its main component is a vacuum vessel, in torus shape, in which the plasma is kept at temperatures of about 100 million degrees for some seconds. The tokamak experiment utilizes real time feed forward and feed back control systems to keep the plasma stable and the fusion device in operation. For the control systems safety and reliability are an obligation. The proposed emulator is one attempt to accomplish these goals. Complex hardware and software systems lack complete testing because of the exorbitant high number of possibilities of internal states. Realtime emulation of the tokamak and its control systems allows many test cycles to be conducted. Utilizing precise boundary conditions the most important internal states of the control software can be tested and verified. With this testing of the software together with redundant systems, fail safe technique and other methods high reliability can be achieved. The use of backup systems for the conduct of the emulation tests allows software development to proceed in parallel to the ASDEX-Upgrade physical experiments.

II. SYSTEM SETUP

Fig. 1: Simplified system setup of the ASDEX-Upgrade Tokamak Control System. (FF=Feed Forward, FB=Feed Back)

Figs. 1 and 2 show the setup for control- and emulator operation. The operational control of the tokamak vessel and plasma is performed by at least two computer systems: one for feed forward and system verification and one for feed back control. To increase reliability these two computers can be assisted by functionally
identical backup systems, using on the fly verification of the control system’s output. At ASDEX-Upgrade only the feedback system is accompanied by a backup, a second feed forward system is available for emulator purposes.

![Emulator setup for real time generation of tokamak data. (FF=Feed Forward, FB=Feed Back).](image)

In emulator mode the feed forward tokamak emulator generates data values which are fed into the backup feed forward control system as if they were real measured data. The data is either preprogrammed or recorded from really executed plasma discharges. Both kinds of data are issued from the feed forward tokamak emulator in real time to test the behaviour of the feed forward control. Besides logical behaviour, deterministic reaction in real time is a key issue for the control system. The same method could be applied for the feedback control system using powerful enough computers for online determination of plasma reactions. Realtime tokamak feedback emulation would require a computer more than $10^4$ faster than the current parallel computers.

In emulation mode the backup computers represent the control system. They are not directly connected to the tokamak and can therefore be used for emulation purposes independently from the tokamak. The main computers are inactive and the backups get their data from the tokamak emulator. Additionally, their output is used as input for the emulator to close the loop.

### III. COMPUTER ARCHITECTURE

At ASDEX-Upgrade all control computers are multi-processor machines built with INMOS transputers [1]. The architecture of these parallel machines is defined by the MULTITOP concept [2, 3, 4]. Fig. 3 shows a symbolic diagram of the architecture of the feed forward tokamak emulator. In the diagram replicated boards are ‘stacked’ for better clearness. Their numbering is indicated in the boxes. The computer consists of three sections (from left to right): the input- the computing- and the output section. Two processor boards, each comprising four transputer memory modules, form the input section. Their task is to protocol the incoming data. In the middle four ‘MCPU’ boards coordinate the input- and output parts. They act as Input Master (IM), Output Master (OM), Central Timer (CT) and Communication Processor (IOP). In the output part four boards with 16 processors are employed. They issue the real time data. The type of boards, used in the in- and output section, is the ‘CPU0’ model of the MULTITOP architecture. The CPU0 boards are accompanied by converter boards which transform fiber optic signals from the periphery to transputer link signals and vice versa.

![Symbolic diagram of the Tokamak data emulator. Data collectors and -dispensers are CPU0 transputer boards.](image)
In figs. 4 and 5 the block diagram of the CPU0 and MCPU is shown. The CPU0 processor board consists of four identical transputer memory modules each equipped with an INMOS Linkadapter [1] which serves as a 'fifth' link as well as a proprietary Transputer Control Bus. The CPU0 is used as 'computing workhorse' while the MCPU acts as a coordinating master. The MCPU processor board therefore has 16 fifth links to control up to 16 transputer memory modules on CPU0s and a processor with large dynamic RAM. The MCPU is also capable of being the master of the Transputer Control Bus.

The key method to connect the boards together to a working computer with backplanes is the MULTITOP transputer control bus and the fifth links. Eight boards with 28 processors in total are connected on three different backplanes to form the feed forward tokamak emulator. In fig. 6 its architecture is depicted. In the upper part of fig. 6 the input- and output sections are shown together with their electrical-optical converter boards. The boards are coupled with the middle part in fig. 6 by two sets of fifth links which form an input- and output star of connections. The two stars are controlled by two MCPU: the Input Master IM and the Output Master OM. Additionally, there is a Central Timer board to issue trigger and timer events and an input/output processor IOP both dedicated to synchronize the emulator with the feed forward control. In the lower part of fig. 6 the components of a SUN SPARC VMEbus workstation are shown together with a CRT display.
The emulator is powerful enough to deliver a data value on each output channel every 300 μs, with an absolute time precision of 1 μs, generated only by software. Additional timer- and trigger signals can be produced by the Central Timer processor (CT) with the same resolution and precision. The data values are transmitted via glass fiber links with 10 MBit/s link speed. Since each transputer has four high speed links the output section sums up to 64 channels. On each channel multiple signals are time multiplexed. Several hundreds of signals can therefore be generated in real time. The emulator input channels read data from the backup feed forward controller with a speed of up to 20 μs for each 3 byte value. The data is marked with a timestamp at the moment of inputting for later analysis. A convenient user interface is implemented on the VMEbus workstation by using X11 window software.

IV. THEORY OF OPERATION

Each projected plasma discharge in the tokamak vessel is planned and defined at ASDEX-Upgrade in a formal manner. Yet all plasma and tokamak parameters which can be preset are summarized and stored in a "shot program". During the execution of the projected discharge the shot program is interpreted by the operation software of the control system. Predefined data values are issued at predefined times to feed forward control plasma and tokamak. Additionally, previously defined control program segments for discharges can be automatically selected by the operation software during the discharge execution. This allows a flexible response to the behaviour of the plasma with a systemwide coherent reaction. The entering and exiting of the different segments is done by online checking physical conditions, such as plasma temperatures etc. The shot program contains some hundreds of signal vectors, their associated time bases as well as the program segment branch conditions. In the emulator each input condition can be simulated and the pre-programmed reaction traced to prove the correctness. A semantical test of a shot program is possible. For a semantical test not only the control system has to be set up by the shot program but also the data values to be generated by the emulator have to be defined. The data values are called "trajectories" since they are functions in time with non aequidistant time intervals. The definition of the trajectories is accomplished in the emulator shot program devoted to emulator and control system. The data values are defined by a simple proprietary language, for which a compiler was made, not by entering each value manually. In this trajectory definition language constant values, ramp up and ramp down functions, time intervals and conditions can be defined by keywords. The simulation itself is divided into
several parts called segments. Each segment is classified in the trajectory definition language by the following parameters: 1) name of the segment 2) set of conditions 3) time range each condition has to be checked 4) maximum time for staying in the segment (watchdog) 5) name of the segment that has to follow if a condition holds 6) default segment after time out 7) set of trajectories that have to be issued as well as their timebases. Each branch condition is formed from a comparison of input value with a specified comparison value. Boolean expressions of conditions can also be specified. Trajectory values can be defined by constant values, a vector of constant values, or by a set of predefined functions (ramp up/ ramp down) with a fixed time interval between each value. Due to the fact that an emulator shot program contains various scenarios to which the shot program execution can be branched depending on the inputs the sequence of segments is not a priori known. As a consequence it is also not known when which values are issued. Therefore, each output value is protocolled with a time stamp as well as name and time of each entered segment is recorded. After a simulation a software tool called 'CDUMP Analyzer' can be used to read the protocol files and to display them in a user friendly manner. The CDUMP tool is written in C and run on the workstation while the control system software is coded in the process oriented language occam [5].

V. RESULTS

The emulator has shown its value in the start up phase of the control system when the tokamak had to be put into operation for the very first time. It was used to test the hardware of the control system and its operating software. Significant time savings were realized by using the emulator, despite the fact that both, control system hardware and software, are not standard or commercially available products. Together with the enhanced security aspects intensive testing had to be performed. Because of the fact that the emulator was separate of the main control computers the development of the operating software could be performed without interfering other start up procedures. Furthermore the emulator has shown its capacity in testing shot programmes which have the shortest life cycle compared to control system hardware and operation software. In two other applications the emulator has potential possibilities: In case of a malfunctioning control system the emulator can be programmed by the protocol files of the data the tokamak and the control system have issued. A post mortem analysis then can be performed easily. Secondly, if a tokamak simulation code is implemented on a powerful emulator computer, it can serve as a new tool for the experimental physicists, because the whole control loop is included.

VII. REFERENCES