



EUROfusion

EUROFUSION WPMST1-CP(16) 15449

J. M. Santos et al.

Real-Time reflectometry - an ASDEX Upgrade DCS plugin App for plasma position and shape feedback control

Preprint of Paper to be submitted for publication in
Proceedings of 29th Symposium on Fusion Technology (SOFT
2016)



This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

This document is intended for publication in the open literature. It is made available on the clear understanding that it may not be further circulated and extracts or references may not be published prior to publication of the original when applicable, or without the consent of the Publications Officer, EUROfusion Programme Management Unit, Culham Science Centre, Abingdon, Oxon, OX14 3DB, UK or e-mail Publications.Officer@euro-fusion.org

Enquiries about Copyright and reproduction should be addressed to the Publications Officer, EUROfusion Programme Management Unit, Culham Science Centre, Abingdon, Oxon, OX14 3DB, UK or e-mail Publications.Officer@euro-fusion.org

The contents of this preprint and all other EUROfusion Preprints, Reports and Conference Papers are available to view online free at <http://www.euro-fusionscipub.org>. This site has full search facilities and e-mail alert options. In the JET specific papers the diagrams contained within the PDFs on this site are hyperlinked

Real-Time reflectometry - an ASDEX Upgrade DCS plugin App for plasma position and shape feedback control

J. M. Santos^a, L. Guimarães^a, C. Rapson^b, G. Santos^a, A. Silva^a, W. Treutterer^b, M. Zilker^b, ASDEX Upgrade Team^b, EUROfusion MST1 Team^c

^aInstituto de Plasmas e Fusão Nuclear, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal

^bMax-Planck-Institut für Plasmaphysik, D-85748 Garching, Germany

^cSee <http://www.euro-fusionscipub.org/mst1>

Abstract

O-mode microwave reflectometry will be used, on ITER and foreseeably on DEMO, to complement the standard magnetic diagnostics for plasma position control. With the preliminary design of ITER's plasma position reflectometers (PPR) presently underway, it is of the utmost interest to test beforehand all possible aspects of this future control application. ASDEX Upgrade (AUG) is the best suited experimental facility on which such tests can be performed. It features a modern, modular and easily adaptable control system, and the only O-mode reflectometry setup capable of probing the plasma at two of the four lines of sight of ITER's PPR (gaps g_3 and g_4). After the first successful demonstration of plasma position control using reflectometry [1], the diagnostic's hardware was updated to acquire a higher number of signals and to improve its real time (RT) data-processing capabilities. Meanwhile, the system's software was rewritten to implement a pipelined architecture to improve its performance and deterministic behavior. The last stage of this pipeline, used to calculate the relevant control parameters, and synchronize and communicate with the discharge control system (DCS), now uses the new DCS software framework, appearing to the control infrastructure as a modular plug-in RT diagnostic App Process. Herein are discussed the adopted synchronization strategies as well as the gains obtained with this new software implementation, namely in terms of performance, fault tolerance, and measurement rate. Experimental data from control discharges is presented to assess the system's operational performance.

Keywords: Real-time reflectometry, Real-time diagnostics, Control system framework

1. Introduction

The operation of future reactor grade fusion tokamaks, such as ITER and DEMO, involves design and engineering challenges that are presently the focus of intense R&D. Among them is the extremely complex task of controlling plasma parameters, relevant for the creation and maintenance of a performant fusion plasma, such as the position of the plasma column inside the tokamak fusion chamber. Avoiding that the plasma impinges the inner vessel walls is essential to allow the heating systems to increase its temperature to several million degrees during the ramp up phase and to prevent destructive disruptions during the steady state full-bore operation. On ITER, O-mode reflectometry will play a supplementary role in providing, at several lines of sight, plasma gap information (plasma to inner wall gaps) to the plasma position and shape controllers. ITER's plasma position reflectometers (PPR) will also have a backup contribution providing measurements of the edge profile as well as of the ELM density transients.

This alternative control scheme, first demonstrated on AUG [1], is being further improved to provide feedback to the preliminary design of ITER's PPR. The improvements described in [2] were recently commissioned to produce a second plasma position demonstration using AUG's two equatorial reflectometers, probing the tokamak high (HFS) and low field sides (LFS).

This upgrade aimed mainly at increasing the system RT measurement rate (4x) and further mimicking ITER's PPR foreseen operation mode.

The calculation of the edge density profile involves probing the plasma simultaneously with a multi-channel reflectometer [3] to cover the measured density range. Additionally, bursts of several consecutive microwave frequency sweeps are used to improve the detected interference signals' SNR and to produce an averaged burst profile []. Fig. 1 shows schematically the acquisition of bursts, B_i , of M sweep data frames, S_j , for a single channel/microwave band. In table 1 are condensed the typical values used in the AUG PPR experiments and the ones currently under consideration for an ITER PPR low and high specification configuration. In the design of PPR RT systems the main control driven requirements are the rate at which density profiles and plasma gaps have to be produced and the total latency involved on their production and delivery to the DCS. In this respect, AUG position control requires a measurement rate, $T_{B_RT} = 1\text{ms}$, 10x faster than the one required for ITER. On ITER, however, requirements for density transient studies demand an effective burst acquisition rate of $T_B = 100\mu\text{s}$. In order to develop and test software and hardware solutions for such scenarios, the AUG PPR burst measurement rate was raised to $T_B = 250\mu\text{s}$, maintaining $T_{B_RT} = 1\text{ms}$.

AUG's RT reflectometry system [2] is used to acquire and

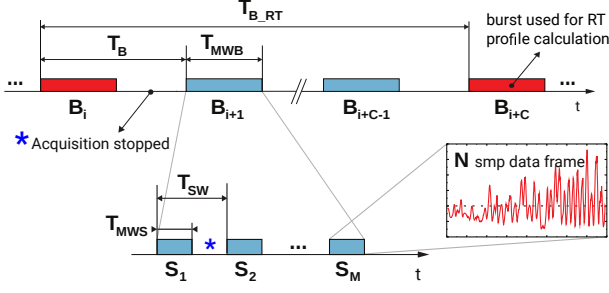


Figure 1: Schematic of reflectometry measurement acquisition timing.

	AUG PPR		ITER PPR (single gap)		units
	HFS+LFS	low spec	high spec		
N	1024	1024	2048		smp.
M	4	4	4		sweeps
C	4	100	100		bursts
T_{B_RT}	1	10	10		ms
T_{MWS}	25	10	5		μ s
T_{SW}	35	15	10		μ s
T_{MWB}	130	55	35		μ s
T_B	250	100	100		μ s
Acquisition system characteristics					
#chan.	16		8		
res.	12		12-16		bit/smp
f_{smp}	40	100	400		MSPS
avg DBW	512	640	1280		MB/s

Table 1: Typical PPR timing and system related values.

process simultaneously data from two PPRs, producing HFS and LFS plasma gaps, and acquiring and handling data from 16 channels (of which only 8 are used for the RT profile calculations). Due to restrictions imposed by the reflectometry hardware, microwave sources can only be swept every $T_{SW} = 35 \mu$ s ($T_{MSW} = 25 \mu$ s actual sweep + 10μ s settling time). To produce a single profile, a burst of $M = 4$ sweeps is used, rising the time required to acquire the full burst to $T_{MWB} = 130 \mu$ s. By using on ITER up-to-date microwave reflectometer hardware these timings can conservatively be reduced down to $T_{MWB} = 35 \mu$ s, easily satisfying the $T_B = 100 \mu$ s requirement.

In spite of acquiring data at burst rates $2.5\times$ slower than the ITER's proposed configurations, the AUG's system is actually handling a similar average total data throughput, 512 MB/s vs 640 MB/s (ITER low spec.), whilst calculating twice as many RT profiles and respective plasma gaps per measured burst. Hence, this setup provides excellent means for the development and live test of a reliable and hardened RT software solutions.

2. RT Reflectometry Diagnostic Software

In its present form, AUG's PPR acquisition and RT data processing system [2] was built around a dual node NUMA server and two custom built ADC boards [4]. Generically, each node is used to acquire and process data from one the two HFS and LFS

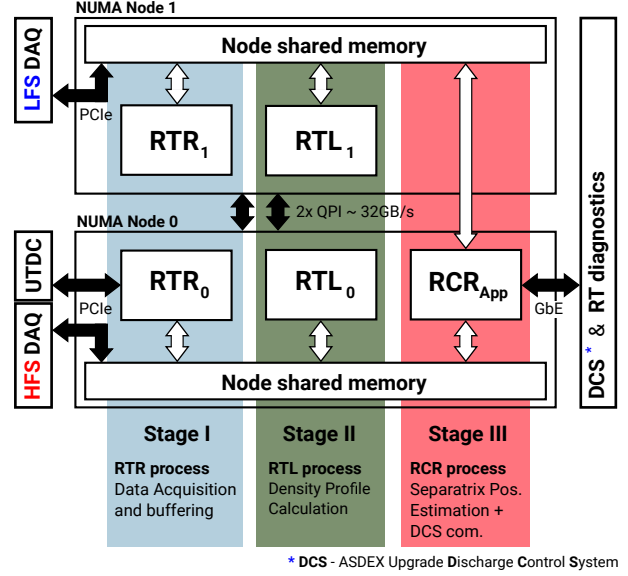


Figure 2: Schematic of the NUMA node mapped software pipeline.

reflectometers. A TDC timing board [5] is used to trigger and timestamp the acquisition of sweeps in the experiment's common timebase. This system is connected to the tokamak discharge control system (DCS) via a Gigabit ethernet connection. Fig. 2 shows schematically the main system components and interconnections as well as the main steps involved in the local software processing. All the large shared memory blocks used by the diagnostic software (≈ 8 GB per node) are preallocated and locked (to avoid page swaps) in each node's local memory at boot time for performance optimization reasons. During the daily discharge cycle, processes simply bind to the relevant blocks and are responsible for their content status and cleanup.

2.1. Pipelined software architecture

To achieve the aimed high rate of RT measurements, a software pipelined approach was implemented. The benefits of isolating the main acquisition, calculation and measurement delivery steps into separate pipeline stage processes are manifold. First, these stages become self contained entities that can generate their own data storage files and be run incrementally providing: a) just raw data acquisition for offline processing (stage I), b) data acquisition + online density profile calculation (stages I+II), or c) data acquisition + density profile calculation + separatrix gap estimation and DCS communication (stage I+II+III). The second benefit resides in the ability to introduce changes in any of the stages without disrupting the tested functionality of the others, as long as the same synchronization and data sharing protocols are maintained. Apart from improving the software maintainability, this separation allows for an easier software optimization, and for a more fine tuned hardware allocation and mapping, namely of the individual NUMA nodes and segregated CPU core sets. Finally, if in the future faster measurement rates are required, dividing and reconfiguring one or more of the existing self-contained pipeline stages is a relatively straightforward procedure. If this change in software

proves not to be enough to guarantee a faster cycle, porting the software pipeline to a new NUMA server with an higher CPU core count (or higher internal bandwidth) should be an easy and risk free task.

2.2. Software pipeline functionality

The first stage process, RTR, is basically a loop polling a memory buffer onto which the ADC boards upload a burst of data using a DMA transfer. As seen in [4], these transfers (1.27 GB/s) overlap in time with the data acquisition of the burst sweeps, and are programmed to end $\approx 1-2 \mu\text{s}$ after the last sample of the burst is acquired. The impact of the acquisition phase on the total latency is thus reduced to an absolute minimum. When a new DMA transfer is available, RTR threads copy the new burst data blocks onto a main raw data buffer in their own node's shared memories, and increment their stage's acquired burst counter. Node zero's thread, after this common phase, still reads and stores the corresponding burst timestamps from the UTDC board, terminating later than node one's.

As soon as RTR threads end the storage of a new burst data block, in each node's main raw data shared memory buffers, the second stage RTL threads immediatly start calculating the corresponding profiles. When this calculation is finished, each thread's last computed profile counter is updated. Four CPU cores are allocated to RTL threads in each NUMA node.

The last pipeline stage, RCR, is implemented using the DCS App framework and runs in a single segregated core in node zero. RCR loops waiting for newly calculated profiles to be available, to produce both HFS&LFS control gap estimates. To produce these estimates, RCR also uses the more up-to-date line average density value, obtained by the framework from the DCS. Even if reflectometry data is available at faster rates, RCR only feeds the DCS with reflectometry control data every $T_{B_RT} = 1 \text{ ms}$, DCS's master control cycle period.

2.3. Integration of RCR in the DCS framework

DCS was conceived from the outset as a distributed control system [6]. More recently, the capability to integrate real-time diagnostics was added [7]. RCR represents the first implementation of a new interface, which offers more functionality to real-time diagnostics. As before, each diagnostic is an independent module running on its own hardware. The interface to non-real-time services such as the parameters server [8] remains the same, including the configuration of the real-time network for each shot, based on a "publish and subscribe" model. RCR uses the new C++ diagnostic interface that offers simple implementation of core DCS functionality, inherited from the software framework existing classes. Particularly useful in this use case were the interface to the TDC, and the synchronization of the RCR algorithm with new interferometry data (which is in turn synchronized to the DCS cycle).

2.4. Pipeline synchronization and fault tolerance/recovery

The characteristics of the used ADCs and the flexibility of the DCS control infrastructure allowed the implementation of a set

of simple, yet effective, pipeline synchronization and fault tolerance/recovery mechanisms, namely: i) the ADC boards ability to tag each acquired data burst with an order index and internal timestamp [4], unequivocally related to the experiment timestamps generated by the TDC; ii) the DCS native support for data-driven workflows [9], not requiring the diagnostic to be hard-synced to the DCS's master control cycle.

Pipeline stage synchronization happens at two distinct phases: i) discharge loop, and ii) burst acquisition/measurement calculation loop. At the beginning and end of each discharge cycle (every $\approx 20-30$ minute), all stage processes synchronize using named counting semaphores. For improved performance, inter-stage synchronization in the runtime-critical burst handling loops is implemented using the shared memory counters described in 2.2. These counters are only incremented by the previous stage, and are only relevant to the next in-chain for synchronization purposes.

RTR polls on the ADC's upload DMA buffer, using the burst index counter inside the uploaded data block to store it in its corresponding place in the shared memory raw data buffer, and to update the thread's burst acquisition counter. RTL stage polls on this counter to know what burst to process. After calculating the density profile, RTL sets the calculated profile counter being polled by RCR, the last pipeline stage. As the operating system (OS) used is not a full fledged hard real time OS, but rather a standard Linux distribution with a RT patched kernel [4, 2], sporadic undesirable system hiccups can occur. Even though the pipeline is running at a higher priority level, if such events make RTR loop skip one of the DMA transfers, or increase the time any of the pipeline stages take to produce their output, the later simply jump directly to the previous stage most recently available data. Because RCR is not hard synced to the DCS master cycle, the DCS is simply temporarily "starved" and no special missing data tagging is necessarily. On the DCS side, if this starvation period is considered too long, the control system might switch to another controller or even initiate a plasma soft landing. Because each stage always knows how many bursts it might have skipped, and the absolute delay between the relevant burst timestamp and the present time (using calibrated CPU internal time counters) more finer grained decision mechanisms can be programmed, so that short skips can nevertheless be processed without huge penalties to the following stage. This higher complexity was not implemented as we aimed at achieving the highest possible measurement rate. In practice, when operating at $250 \mu\text{s}$ the first two stages can always miss 3 consecutive burst without risking DCS starvation (DCS master cycle is 1 ms).

These simple mechanisms guarantee that the pipeline has no deadlock conditions, always automatically recovering from system delays in the fastest possible way.

3. System benchmarking

The experiment's time base, accessible via the local TDC board, was used has a reference to produce the benchmarks shown in this section. This board timestamps every measurement allowing the direct calculation of the delivery latencies to

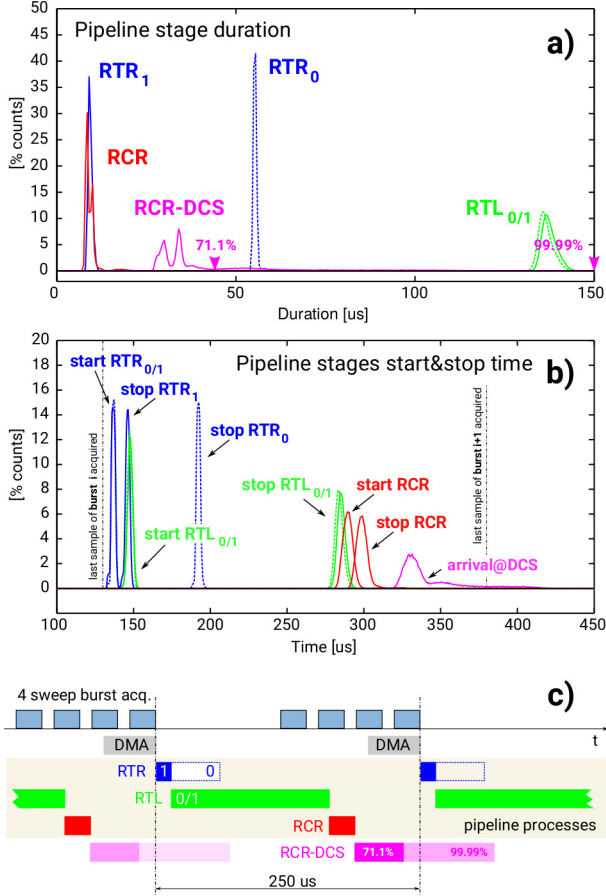


Figure 3: a) Pipeline stage duration, b) measurement cycle pipeline timings, c) Pipeline stage timing diagram. (data from 6 discharges, i.e. 6x40000 bursts)

the DCS, also using the same timebase. Pipeline stage thread benchmarking was performed using the CPUs' internal timestamp counters (after proper calibration with the TDC timer) due to the highly efficient concurrent access to these timers (sub μs latency). The histograms on Fig. 3, obtained in six discharges (6x40000 measurements), show a) the duration of each of the pipeline stages and b) the start and stop times referenced to the acquisition trigger of the first sweep of each burst. Also in the plots, labeled *RCR-DCS*, is the time required to deliver the gap estimates calculated by RCR to the DCS. The corresponding arrival at the DCS curve, Fig. 3.b), characterizes the total latency of the system, found to be $< 450 \mu\text{s}$. The longest pipeline stage is RTL, the density profile calculation stage, whose duration ($< 150 \mu\text{s}$) defines the fastest achievable full pipeline cycle (presently set to $250 \mu\text{s}$). Operating the pipeline with a $150 \mu\text{s}$ cycle period would still be compatible with the $130 \mu\text{s}$ needed to sweep and acquire the 4 sweep bursts (a microwave system requirement). In this configuration, the total average data bandwidth flowing into the server would reach 854 MB/s, between the 1280 and 640 MB/s bandwidths of the high and low specifications for the ITER PPR systems (see table 1).

RTR and RCR are naturally faster due to the limited amount of operations processed inside their burst loops. The calculation of both HFS & LFS gaps from the profiles in RCR, takes as

much time as storing the 64 KB data blocks to main memory in RTR, $\approx 9 \mu\text{s}$. As soon as both RTR threads finish this transfer, RTL threads on both nodes start and end almost simultaneously ($\sim 1\text{-}2 \mu\text{s}$ apart). As explained before, RTR thread running on node zero, RTL_0 , ends $\approx 45 \mu\text{s}$ later than RTL_1 for reading the burst timestamps from the TDC.

As no scatter-gather is implemented on the ADC boards' DMA engines, both boards are actually uploading data to the same DMA lower memory region on NUMA node 0. RTR threads on both nodes concurrently poll on this memory region before copying their 64 KB burst data blocks to their own node's local main shared memory buffer. From Fig. 3.a), RTR₁ loop duration indicates that this concurrent memory access over the inter-node QPI links is performed at an effective 7.1 GB/s data rate. This is several times higher than the effective bandwidth of the PCIe 1.1 x8 interface of the used ADC boards (1.27 GB/s, [4]) and still higher than the effective bandwidth of typical PCIe 3.0 x8 interfaces ([10]).

Fig. 3.c) shows a simplified pipeline timing diagram (using the highest observed latencies). *RCR-DCS* darker color bar represents 71% of measurements delivered to DCS, whilst the lighter bar corresponds to 99.99%. It can be seen that breaking the calculation stage in two parts would easily allow the maximum measurement rate to be lowered well below the $150 \mu\text{s}$ mark if the total burst acquisition cycle could be lower than $140 \mu\text{s}$ ($130 \mu\text{s} + 10 \mu\text{s}$ settling time).

4. Control experiments and system performance

During the 2016 experimental campaign the described system was used to demonstrate plasma position control using both inner (HFS) and outer (LFS) O-mode reflectometers. The RT estimates of the inner, R_{in} , and outer, R_{aus} , separatrix positions were combined to produce a naïve approximation of the geometric plasma radius, $R_{geo} = (R_{in} + R_{aus})/2$, that replaces the corresponding magnetic measurement normally used as the controller input signal.

Fig. 4 shows the main time traces of one of the 4 control discharges performed. The top plot shows the line integrated density (H_1) at the equatorial plane, Deuterium fueling (D), neutral beam (NBI) and ECRH ($ECRH$) heating and plasma current (I_{pa}). During the flat-top ELMy H-mode phase, position control is handed to the reflectometry based controller from $t \approx 2.6$ s until $t \approx 7.6$ s, when the magnetic controller kicks in to perform the plasma ramp down.

During the reflectometry control phase, the geometric radius of the plasma column was programmed (Injected Trajectory trace on third plot of Fig. 4) to swing 1.5 cm (non-symmetrically) around its original position. As can be seen the new controller maintained reflectometry's R_{geo} within $\approx \pm 0.5$ cm of the target trajectory. Reflectometry estimates for R_{in} , R_{aus} , and R_{geo} are coherent with their magnetic counterparts, demonstrating very good precision although with improvable accuracy: ≈ 1.5 cm and ≈ 2 cm offsets to the magnetics at the LFS and HFS, respectively. The corresponding input R_{geo} offset is successfully handled by the position controller when switching to and from reflectometry input at $t \approx 2.6$ s

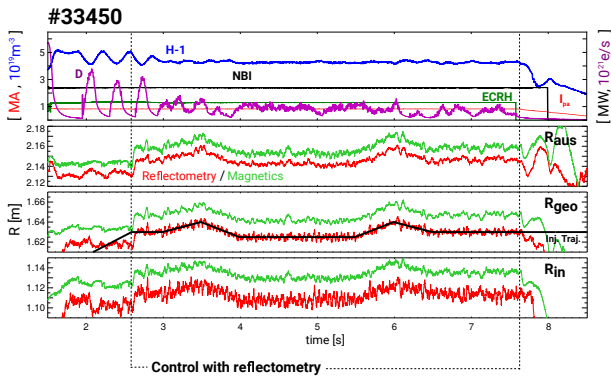


Figure 4: Controlling R_{geo} using HFS&LFS reflectometry profiles (discharge #33450).

and $t \approx 7.6$ s. In all four control discharges (#33448, #33450, #33452 and #33453) the system operated flawlessly during the complete programmed reflectometry control phases. The uninterrupted stream of control R_{geo} estimates, produced every 1 ms, reach the DCS with a total latency always $< 350 \mu\text{s}$.

5. Outlook

The success of these control experiments using reflectometry density profile measurements proved once more that, for ITER, this is a sound alternative or complement to the traditional magnetic based position control. Moreover, it was shown that the recently introduced system upgrades and new software developments not only worked flawlessly but also brought the system one step closer to the fulfillment of ITER's PPR requirements. AUG's control and O-mode reflectometry setup continues to be the ideal test ground for the solutions found during the design phase of ITER's PPR, not only at the algorithmic level but also at the control and diagnostic system levels.

Now that the base diagnostic development and DCS integration have been reached, many areas for improvement have already been identified: i) interventions in the ≈ 20 year old reflectometer system to improve signal quality will decrease the complexity of algorithms required to produce reliable and more accurate RT measurements; ii) the same upgrades will potentially allow the microwave sources to be swept faster, enabling ITER measurement rates of $T_B \leq 100 \mu\text{s}$ and triggering pipeline stage (RTL) revisions to increase the system throughput; iii) further integration of the middle pipeline stages in the control App paradigm in case multithreading support (OMP, pthreads, etc) becomes available in the DCS framework; iv) finally, if faster reflectometry measurements prove to be useful on AUG for RT density transient signaling (ELMs, L-H transitions, etc), lower latency connections such as reflective memories [7] will have to be implemented.

Acknowledgments

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under

grant agreement No 633053. IST activities also received financial support from Fundação para a Ciência e Tecnologia through project UID/FIS/50010/2013 and grant SFRH/BD/87738/2012. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

- [1] J. Santos, L. Guimarães, M. Zilker, W. Treutterer, M. Manso, Reflectometry-based plasma position feedback control demonstration at ASDEX Upgrade, Nucl. Fusion (3) 032003. doi:10.1088/0029-5515/52/3/032003.
- [2] J. M. Santos, G. S. Santos, M. Zilker, L. Guimarães, C. Rapson, W. Treutterer, B. Gonçalves, Enhancement of the ASDEX upgrade real-time plasma position reflectometry diagnostic, IEEE Trans. Nucl. Sci. (3) 1195–1200. doi:10.1109/TNS.2015.2393692.
- [3] A. Silva, M. E. Manso, L. Cupido, M. Albrecht, F. Serra, P. Varela, J. Santos, S. Vergamota, F. Eusebio, J. Fernandes, T. Grossmann, A. Kallenbach, B. Kurzan, C. Loureiro, L. Meneses, I. Nunes, F. Silva, W. S. A. U. Team, Ultrafast broadband frequency modulation of a continuous wave reflectometry system to measure density profiles on ASDEX Upgrade, Rev. Sci. Instrum. (12) 4138–4145. doi:10.1063/1.1147517.
- [4] J. Santos, M. Zilker, L. Guimarães, W. Treutterer, C. Amador, M. Manso, the ASDEX Upgrade Team, COTS-Based High-Data-Throughput Acquisition System for a Real-Time Reflectometry Diagnostic, IEEE Trans. Nucl. Sci. (4) 1751–1758. doi:10.1109/TNS.2011.2143428.
- [5] A. Lohs, K. Behler, K. Lüddecke, G. Raupp, T. A. U. Team, The {ASDEX} upgrade {UTDC} and {DIO} cards - a family of {PCI/cPCI} devices for real-time {DAQ} under {Solaris}, Fusion Eng. Des. 81 (2006) 1859–1862.
- [6] W. Treutterer, K. Behler, R. Cole, J. Hobirk, M. Jakobi, A. Lohs, K. Lüddecke, G. Neu, G. Raupp, W. Suttrop, D. Zasche, T. Zehetbauer, M. Zilker, The new ASDEX upgrade real-time control and data acquisition system, Fusion Eng. Des. 66-68 (2003) 755–760. doi:10.1016/S0920-3796(03)00298-9.
- [7] W. Treutterer, L. Giannone, K. Lüddecke, G. Neu, G. Raupp, D. Zasche, T. Zehetbauer, Real-time diagnostic integration with the ASDEX upgrade control system, Fusion Eng. Des. (7-11) 1871–1874. doi:10.1016/j.fusengdes.2008.12.026.
- [8] G. Neu, R. Cole, A. Gräter, K. Lüddecke, C. J. Rapson, G. Raupp, W. Treutterer, D. Zasche, T. Zehetbauer, The ASDEX Upgrade Parameter Server, Fusion Eng. Des. 676–679doi:10.1016/j.fusengdes.2015.06.065.
- [9] W. Treutterer, G. Neu, G. Raupp, D. Zasche, T. Zehetbauer, R. Cole, K. Lüddecke, Management of complex data flows in the ASDEX Upgrade plasma control system, Fusion Eng. Des. (12) 2039–2044. doi:10.1016/j.fusengdes.2012.01.038.
- [10] Y.-k. Choi, J. Cong, Z. Fang, Y. Hao, G. Reinman, P. Wei, A quantitative analysis on microarchitectures of modern CPU-FPGA platforms, Proc. 53rd Annu. Conf. - DAC '16 1–6doi:10.1145/2897937.2897972.