EPFL

Lecture #6: Control systems technology for a Tokamak Plant

Cristian Galperti

Swiss Plasma Center

06.02.2025

Summary

- Slow control, plant supervision
- Real time hardware control basics
- Digital processing and control technologies
- Real-time networks
- TCV's digital control system hardware
- Actuators
- Tokamak control systems software aspects
- TCV's digital control system software, applications

Tokamak machines from the control point of view

Galperti Cristian

- Tokamaks are part of the so called "**big-physics**" devices that encompass many technologies
- They run in a "**plant**" that takes 24/7 care of systems like temperature, vacuum, security etc.
- They are still mostly "**pulsed**" devices that have physics operations for often (far) less than 10% of wall-clock time
- During not operation periods control usually is delegated to slow unmanned plant control systems
- During these pulses, **heightened** activity is common and real-time feedback control is mandatory to control and sustain the plasma and to achieve scientific results

Slow control (plant supervision)

EPFL

Slow control system architecture

Galperti Cristian

Keep ALL systems necessary for running all components of the plant (including water flows, air temperatures, servicing requirements all the way to machine shot preparation)

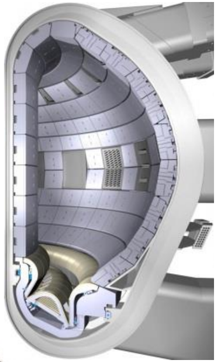
HMIs

↔

Plant “live” database

Usual update rate < 10 Hz

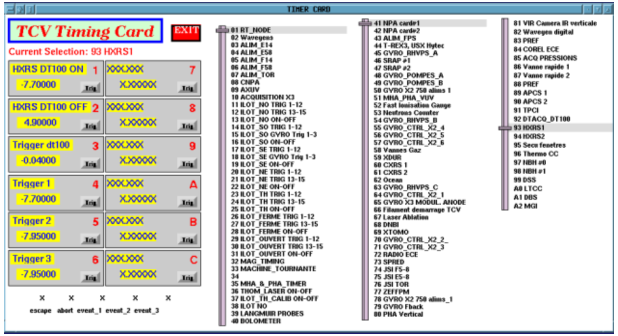
↔

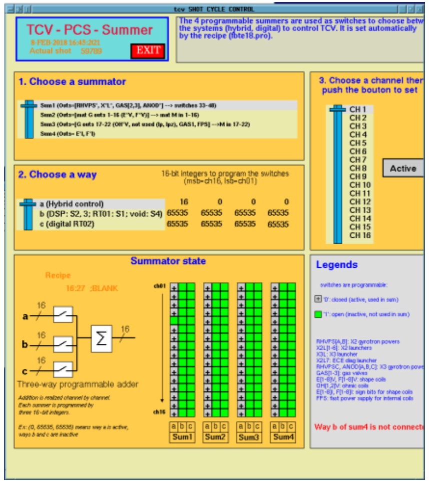


“24/24H” PLANT

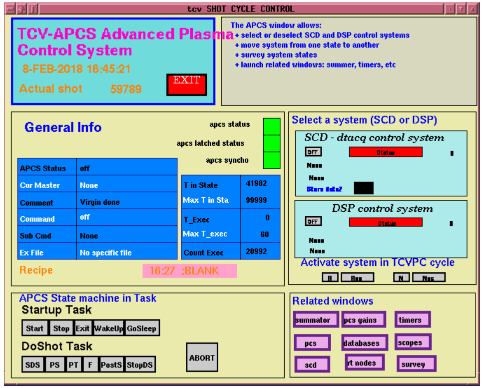
Swiss Plasma Center

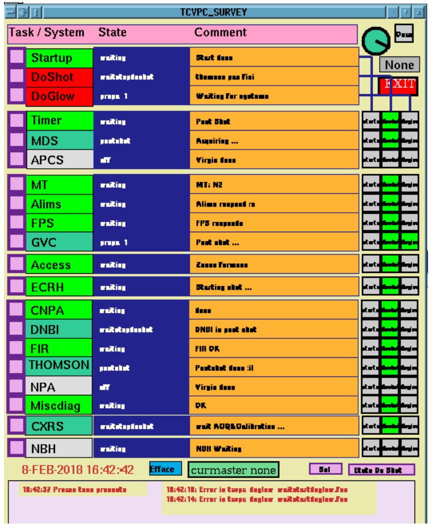
TCV example





VISTA “live” database





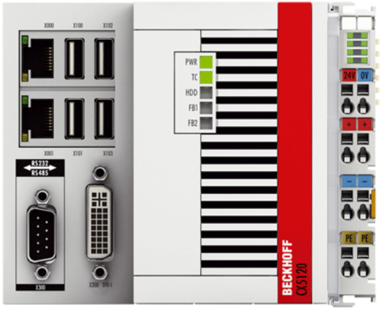
EPFL

OPCUA protocol

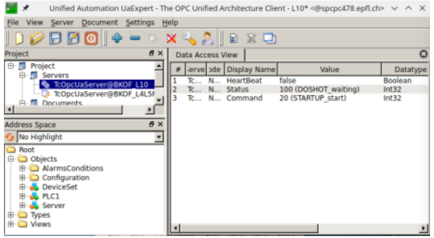
Production live DB client

OPCUA connection

OPCUA server running on a local control system



OPCUA connection



GUI based debug client

OPCUA connection

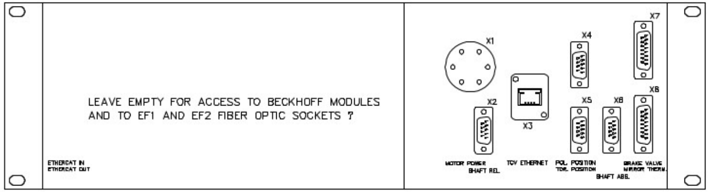
```
from opcua import Client
from opcua import ua

def sendcommandloop(server):
    client = Client(server)
    try:
        client.connect()
        # Command dictionary from the
        #cmddict=slaveint.get_command
        # direct nodeid call method (
        Command=client.get_node("ns=4
        usercmd = 0;
```

Script based debug client

- Vendor and platform independent
- Connection based (server/client model)
- Secured (encryption and authentication)
- Available from many control equipment manufacturer as well as open source developers

https://opcfoundation.org/about/what-is-opc/

A modern integrated subsystem

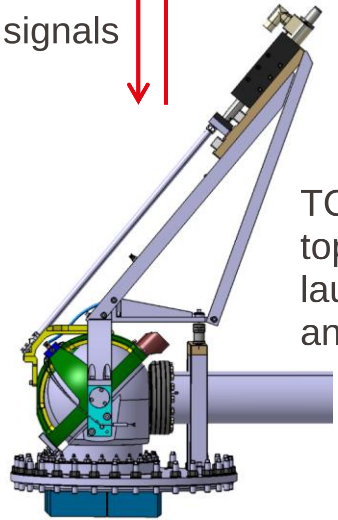


Commands and status exchange via OPCUA protocol and OPCUA/Vista bridge

Vista DB

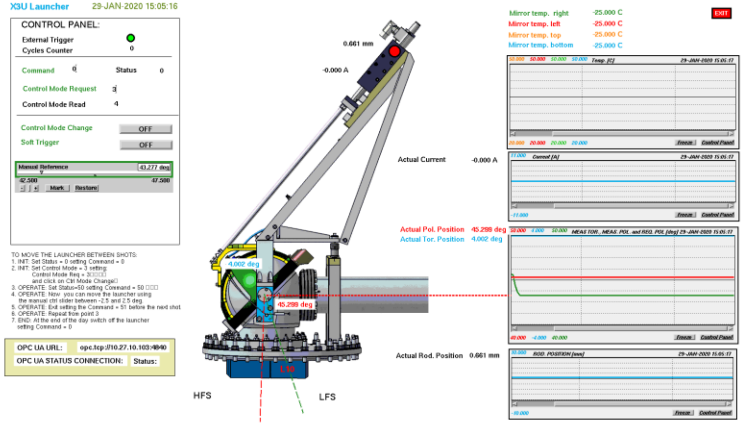
Launcher local controller

Field signals



TCV ECRH top fast-launchers L10 and L11

Launcher control panel



Galperti Cristian

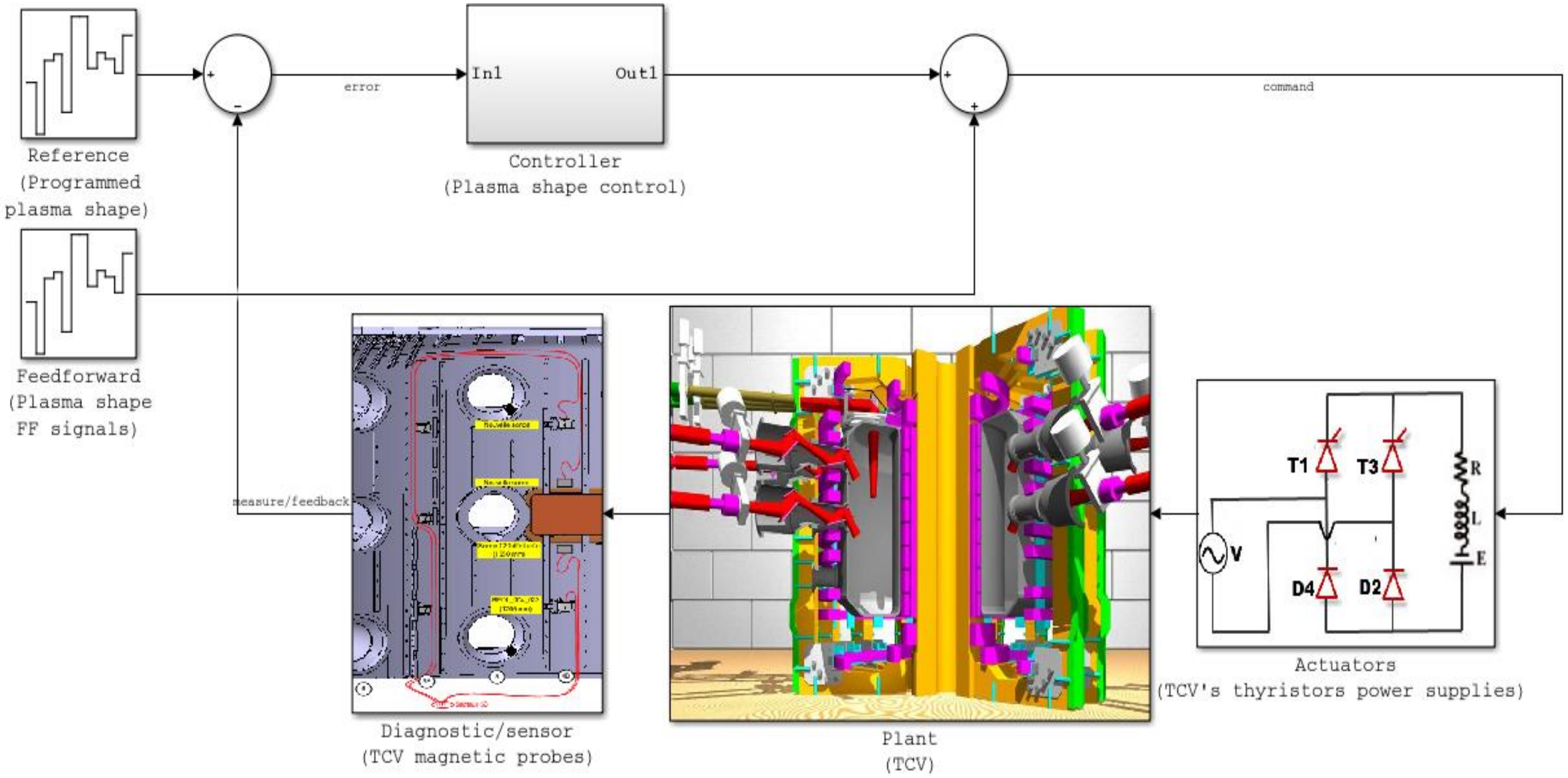
Swiss Plasma Center

Real-time control hardware basics

Swiss
Plasma
Center

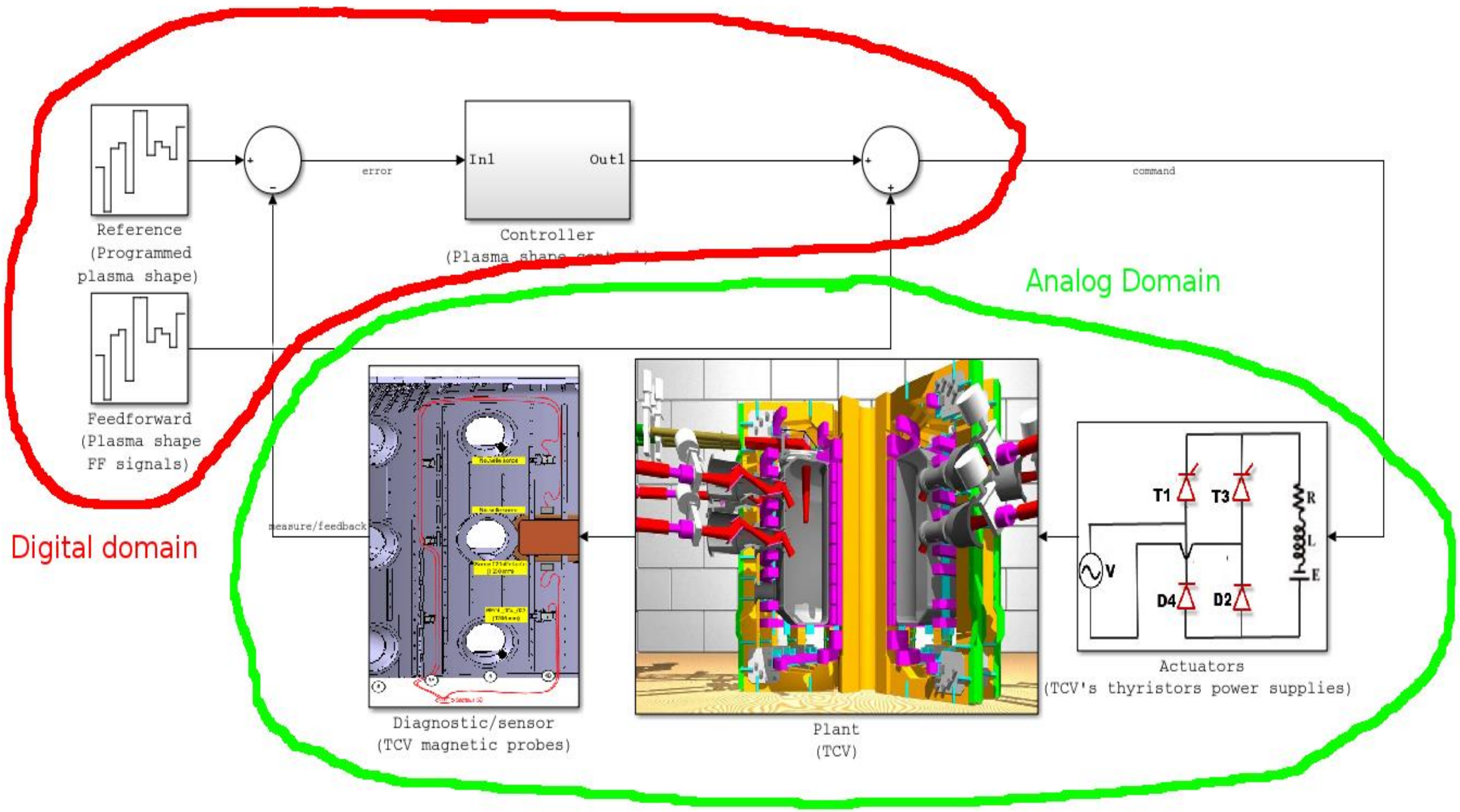
Digital real-time control of tokamaks

Galperti Cristian



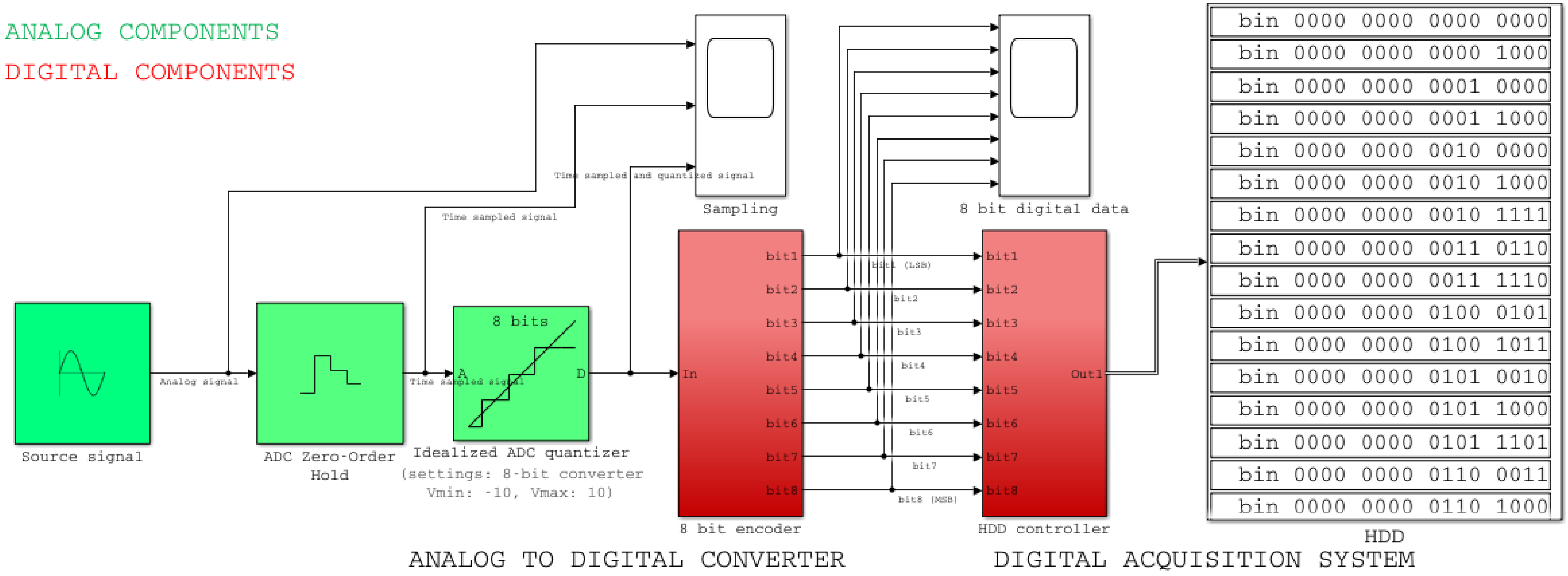
Swiss Plasma Center

Digital real-time control of tokamaks



Galperti Cristian

Analog to digital conversion (1)

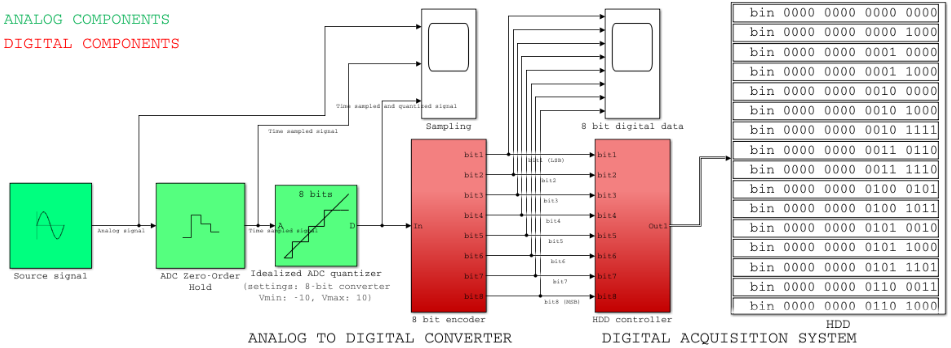


Analog to digital conversion (2)

Galperti Cristian

Common problems and pitfalls

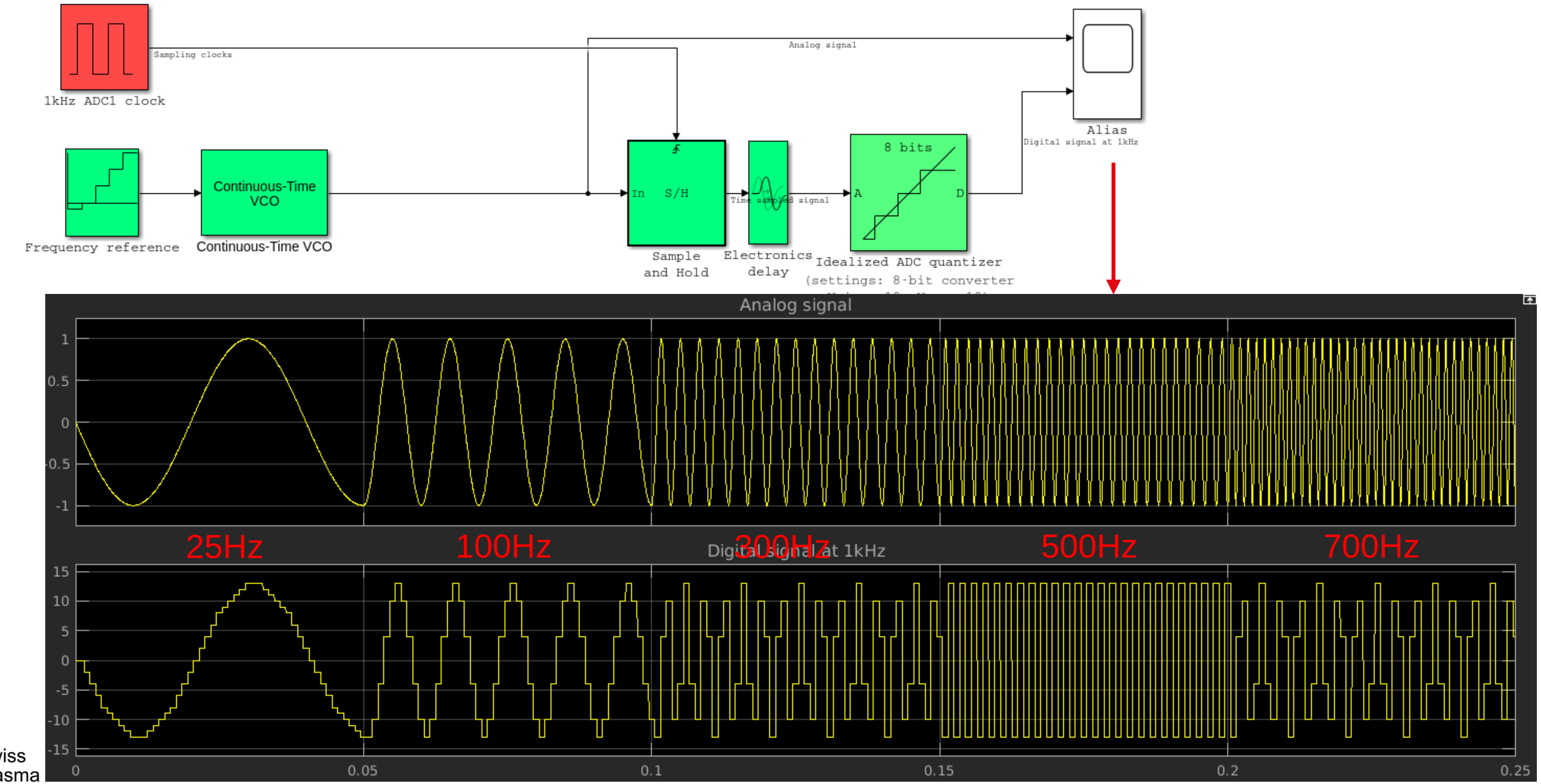
- Wrong input signal level
  - too low: quantization noise -> open loop
  - too high saturation and signal clip -> open loop
  - Need adjustable input amplifiers
- Wrong input signal speed
  - Alias effect and signal and noise spectrum pileup -> poor controller performances
  - Need correct input analog filtering



- Wrong sampler timing
  - Loss of synchronicity with the rest of the plant
  - Wrong storage timebase -> control system resimulation almost impossible

Swiss Plasma Center

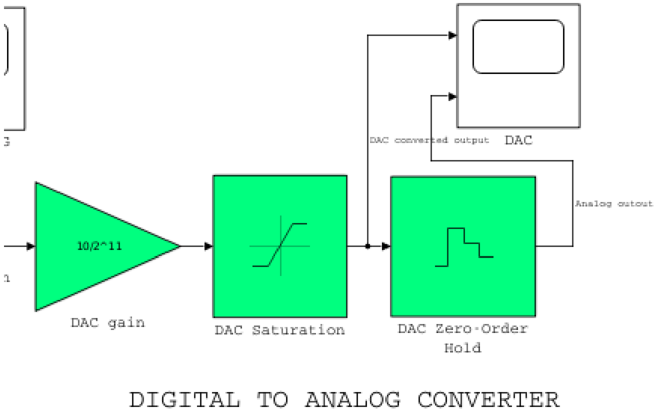
Analog to digital conversion, sample frequency



Digital to analog conversion

Common problems and pitfalls

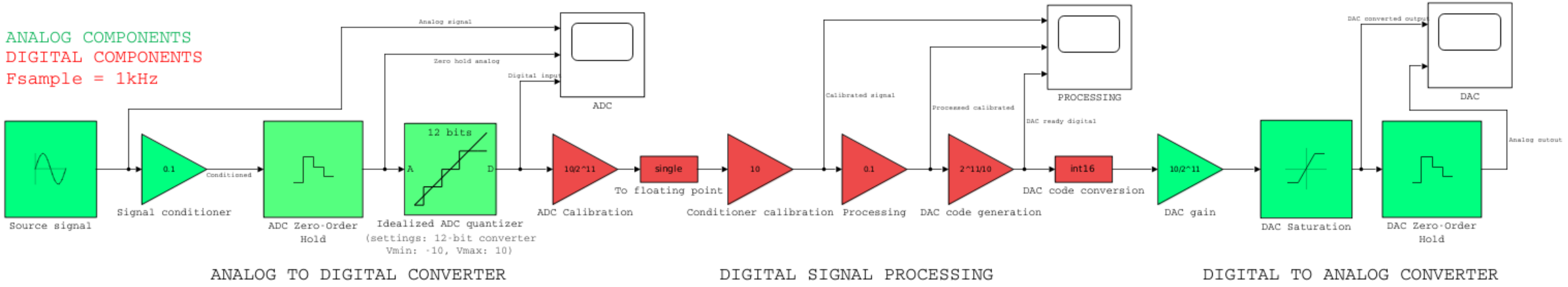
- Wrong output signal level
  - too low: quantization noise -> open loop
  - too high saturation and signal clip -> open loop
  - Need well designed control math and DAC – actuator matching
- Too poor resolution / too high offset
  - Sometimes poorly designed control-actuator chains may lead to too little control resolution / too high offset



Suggested common solution: use direct digital control – actuator interfaces whenever it is possible

A minimal digital control chain

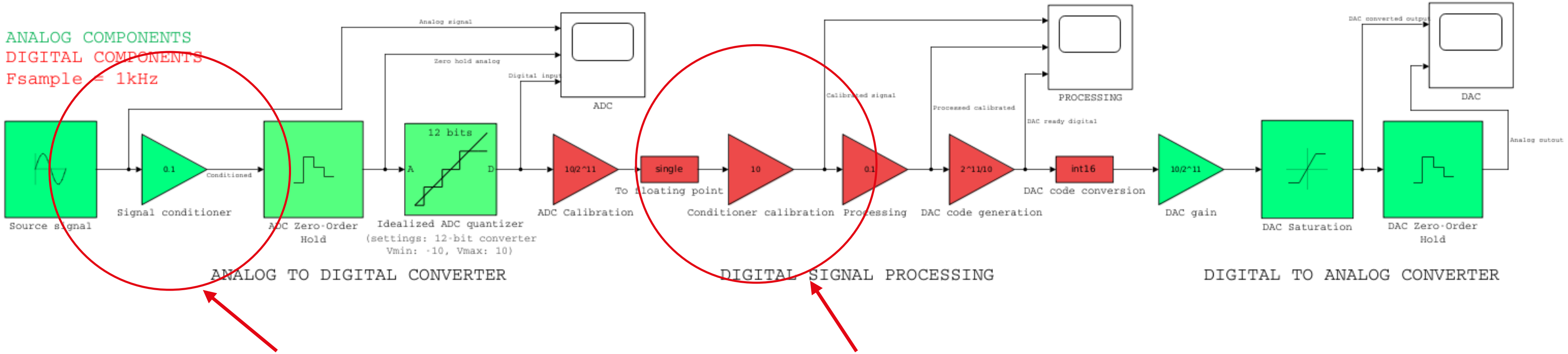
Galperti Cristian



Swiss Plasma Center

A minimal digital control chain

Galperti Cristian



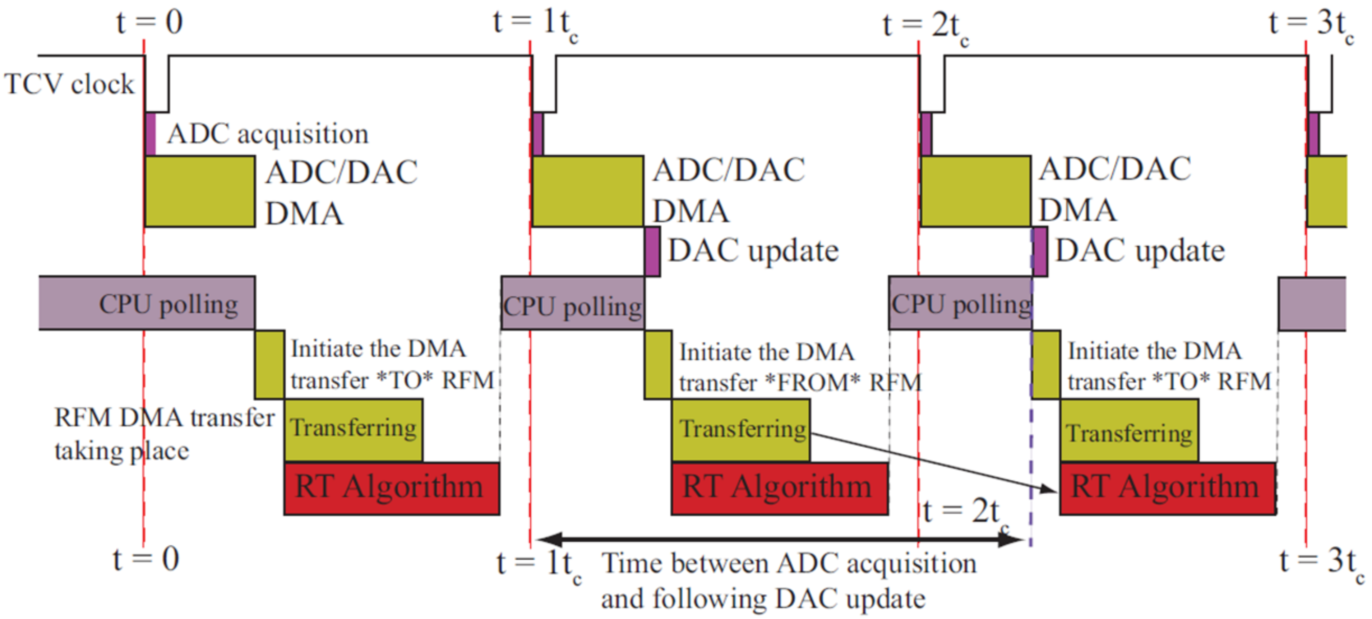
The signal is pre-treated in analog domain before The ADC

In digital we compensate for the analog pre-treatment

Swiss Plasma Center

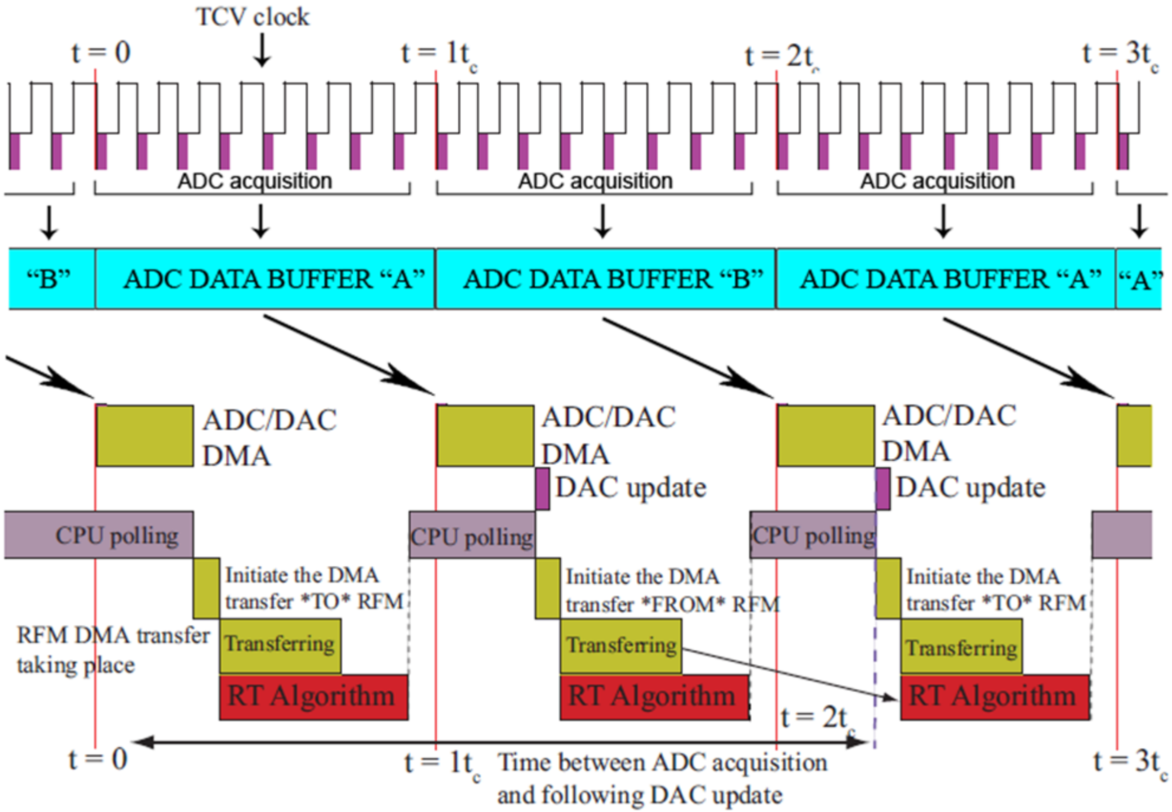
Time chart of a real-time computer

Typically, a real-time digital control hardware cyclically executes the control code triggered by a timing system. Usually the trigger is common with the ADCs (but not mandatory). Multiple concurrent processing and/or data transfers may happen (especially with multi cores CPUs)



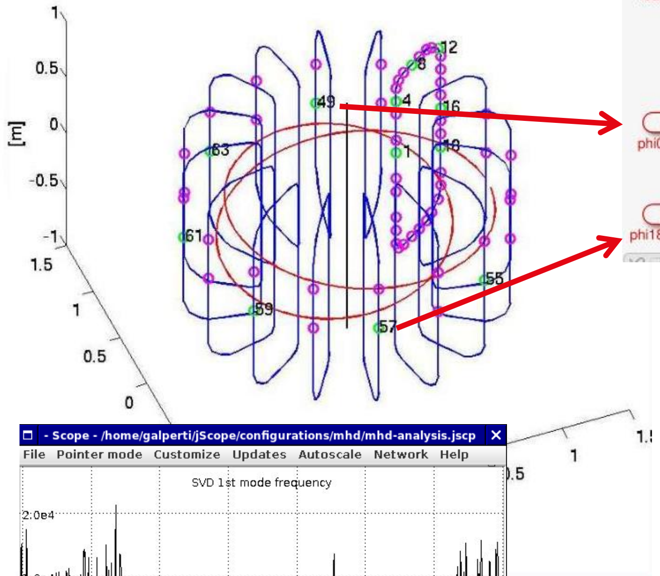
From J. I. Paley et al. Real Time Conference (RT), 2010 17th IEEE-NPSS

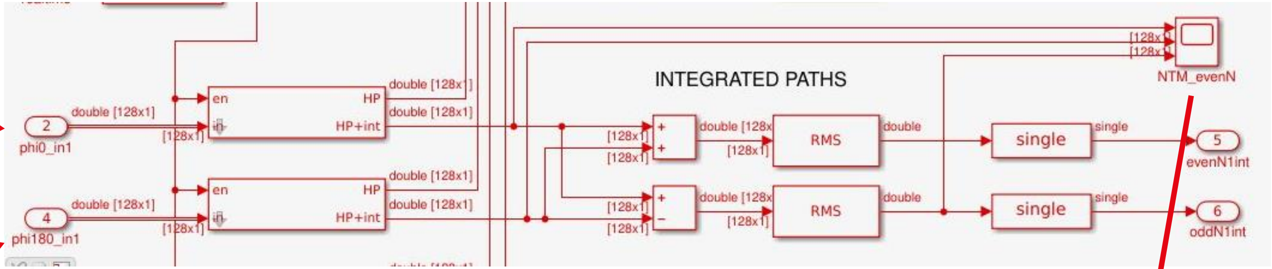
Time chart of a real-time oversampling computer

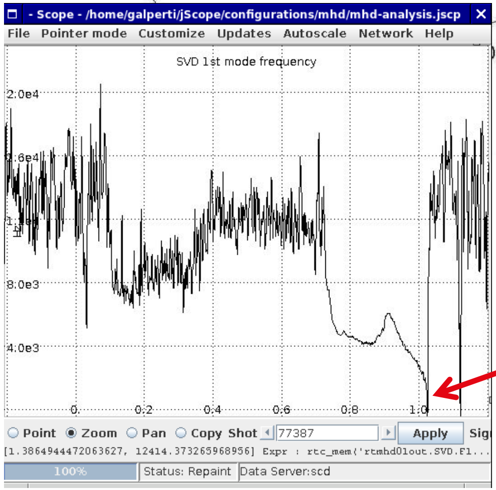


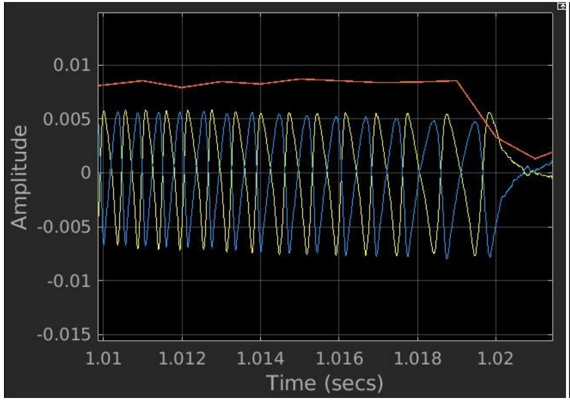
- Basically a (double) buffer is inserted between the ADC digital outputs and the DMA source endpoint.
- ADC data are now tranferred in bursts into the host PC memory. This frees the sampling frequency upper bound.

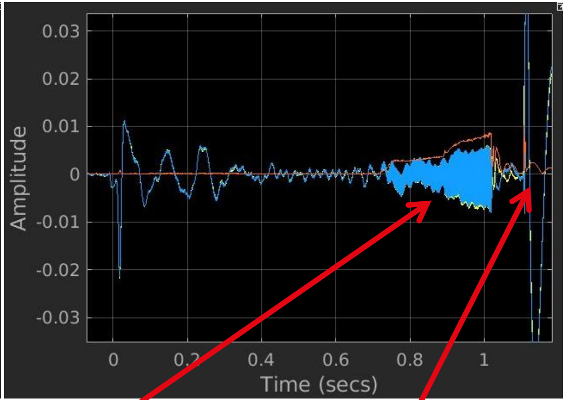
An example, real-time magnetic islands detection











Mode rotating frequency (locking)

Even n mode

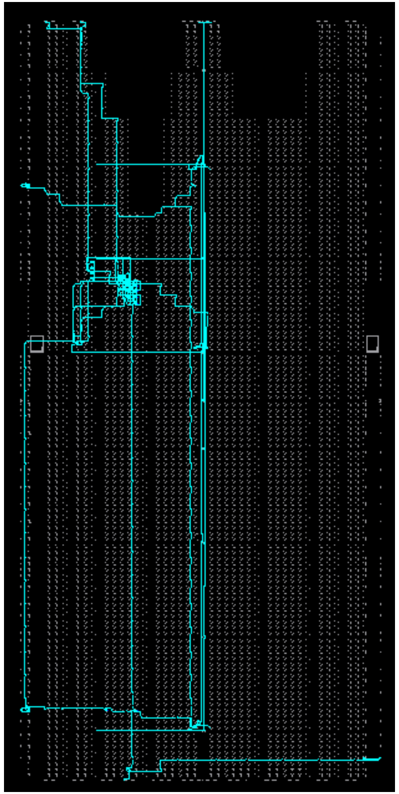
Plasma disruption

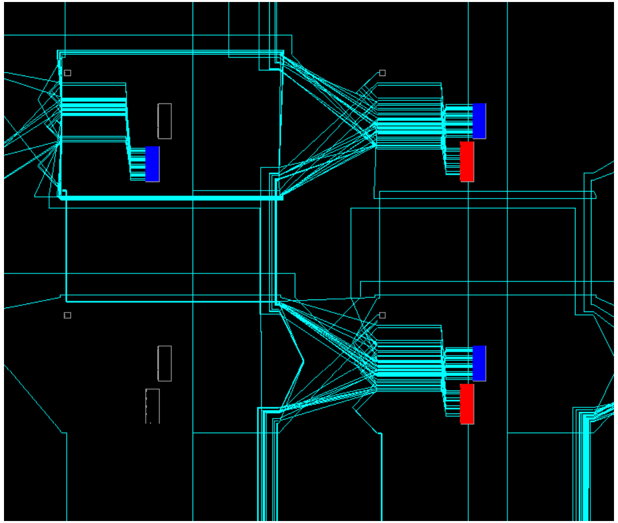
https://doi.org/10.1109/TNS.2017.2711625

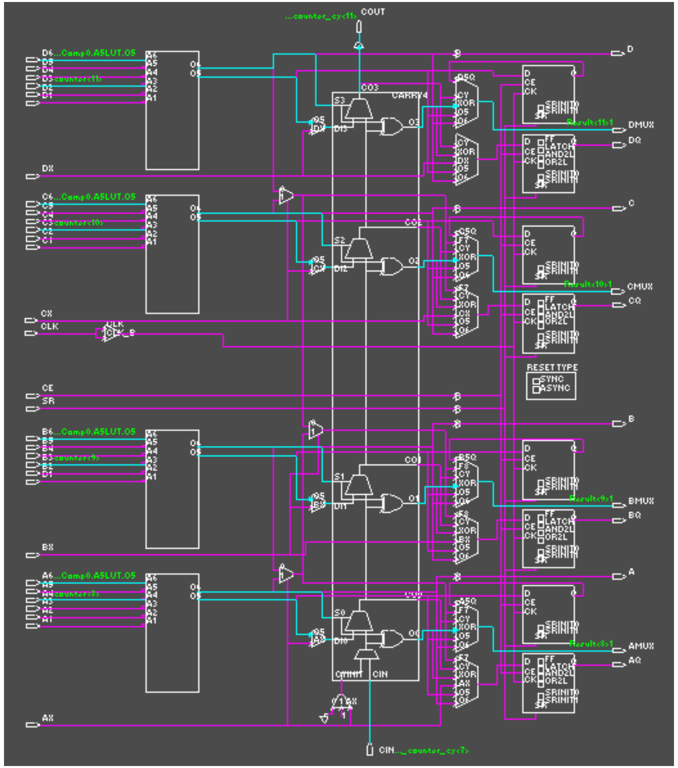
Digital processing and control technologies

FPGAs

Galperti Cristian

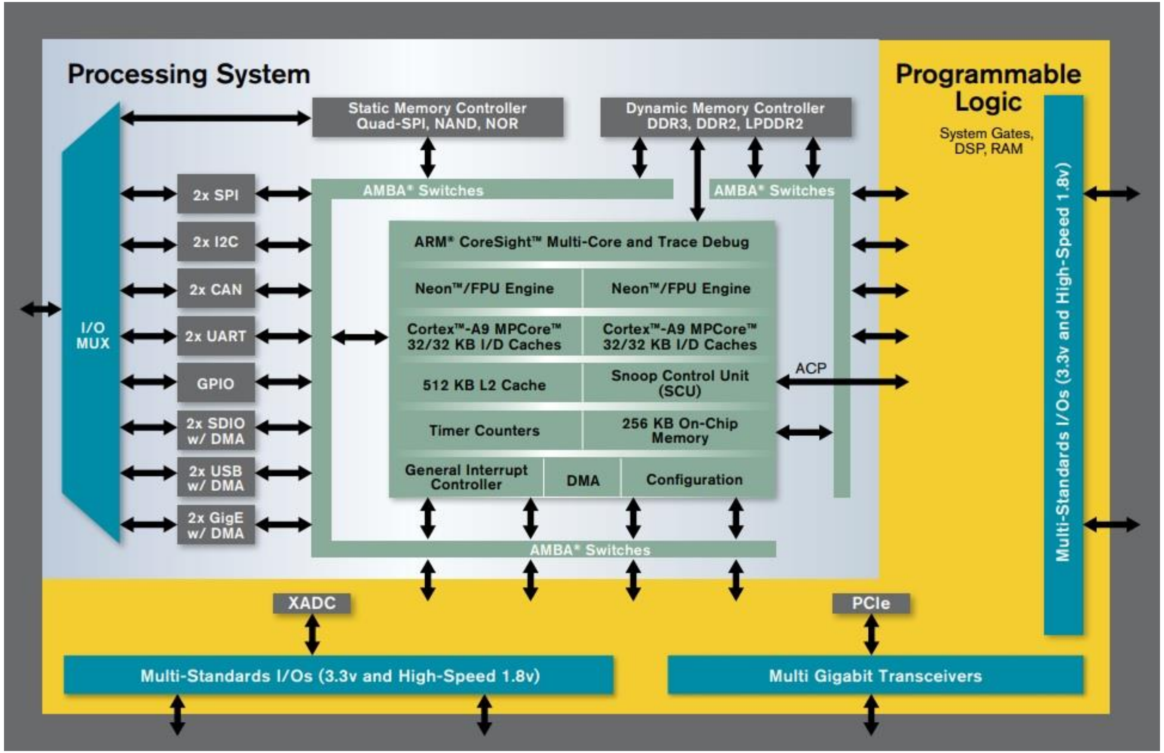






FPGAs (Field Programmable Gate Arrays) chips allow an incredible degree of hardware design flexibility at viable costs. But they are usually quite difficult to program/commission. Here an example of a routed clock DPLL clock converter for TCV (1 MHz in -> 320 MHz internal -> 256 kHz synchronous out)

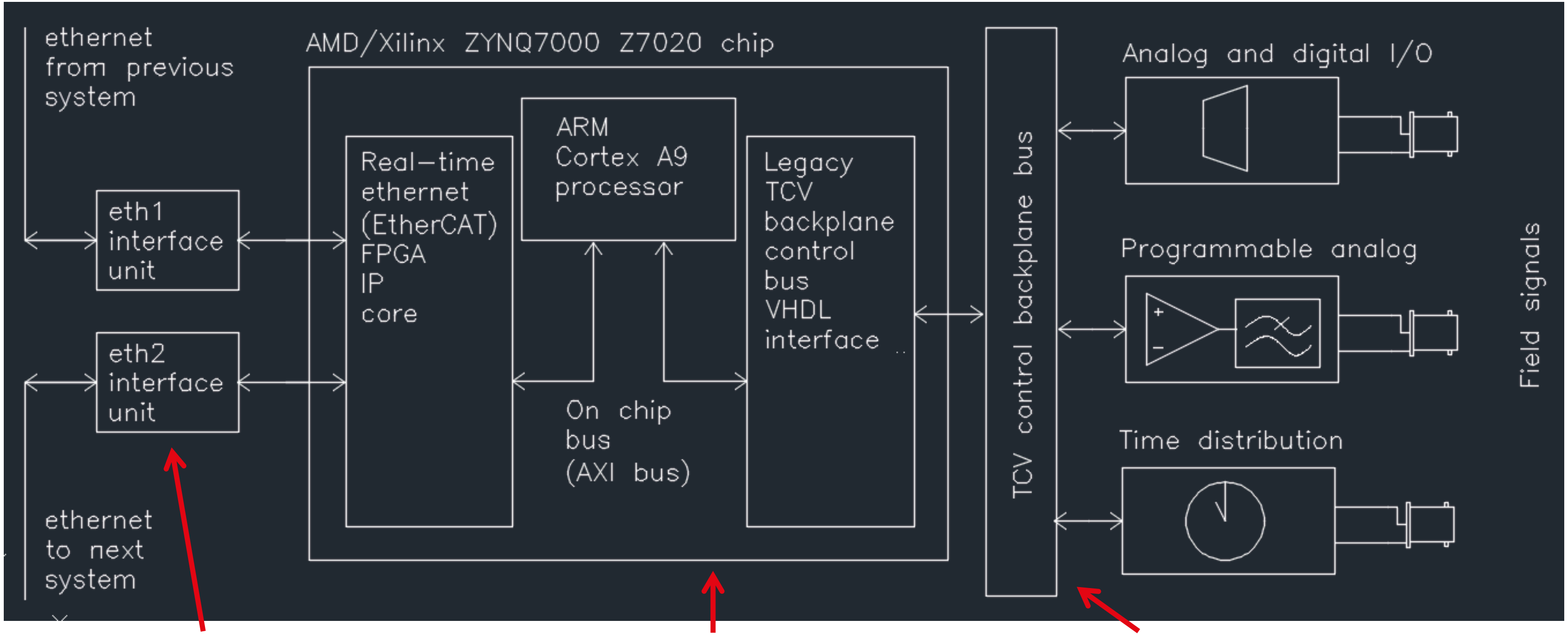
SoC (System on chip)



System on chip combine support for complex code execution (e.g. kernels) given by CPUs to custom circuitry flexibility granted by FPGA on the same chip. The may also embed specific standard I/O modules and/or mainstream bus endpoints like PCIe. They are the primary choice nowadays for high performance control embedded systems, even in fusion.

AMD/Xilinx ZINC7000 SoC
https://www.xilinx.com/products/silicon-devices/soc/zynq-7000.html

SoC, new TCV distributed communication processor



Communication field-bus

- Daisy chainable, EtherCAT based
- Real-time data exchange support
- Precise time distribution support

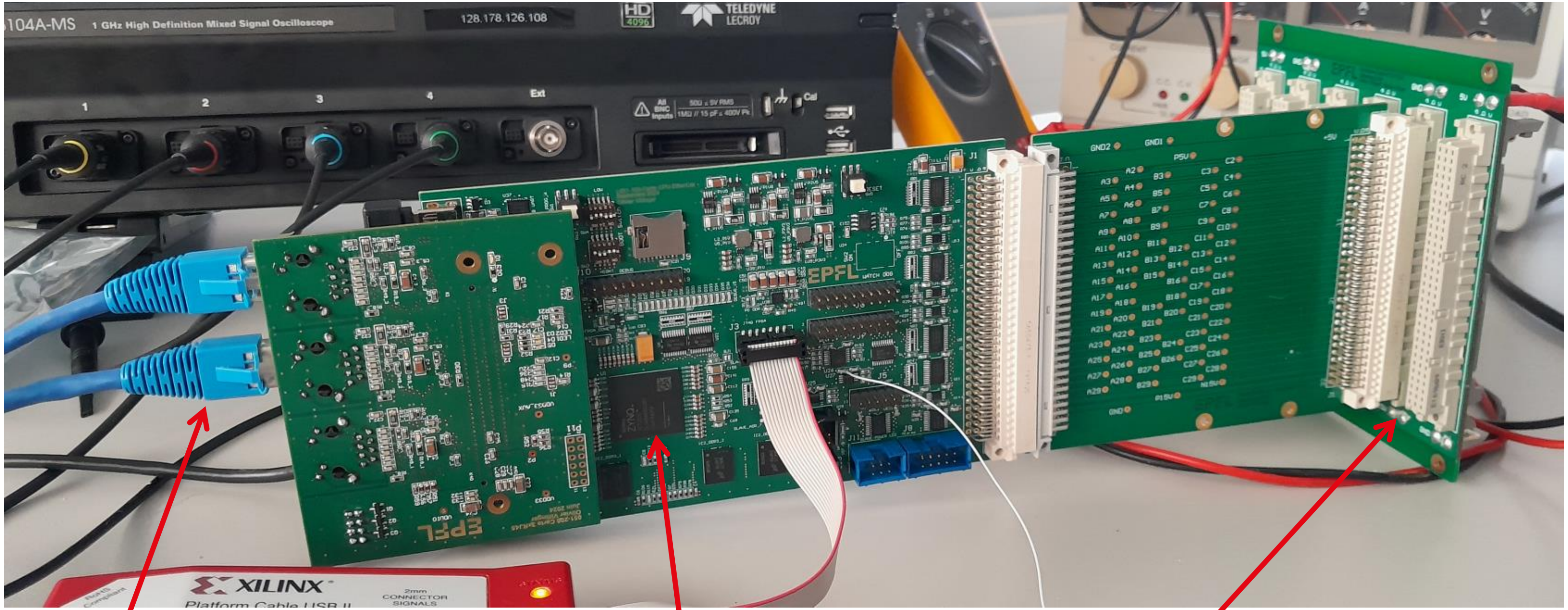
AMD/Xilinx ZYNQ7000 System on Chip

- Highly configurable, future proof architecture
- Real-time network data traffic and time handled in hardware

Legacy TCV control systems backplane

- Supports all already developed control electronics systems (since 30 years)
- Allows integration of new and modern control communication networks into legacy control hardware

SoC, new TCV distributed communication processor



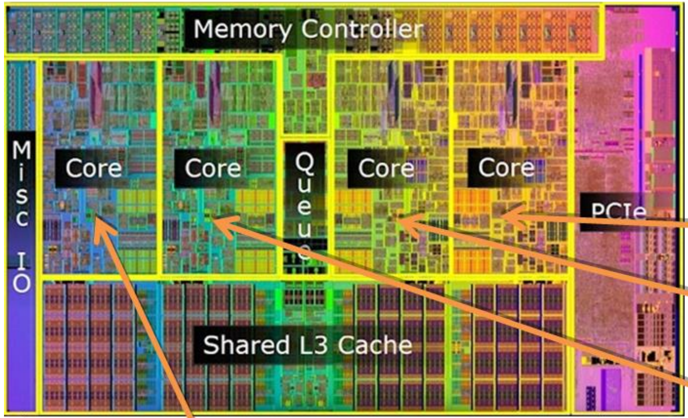
Communication field-bus
- Daisy chainable, EtherCAT based
- Real-time data exchange support
- Precise time distribution support

AMD/Xilinx ZINQ7000 System on Chip
- Highly configurable, future proof architecture
- Real-time network data traffic and time handled in hardware

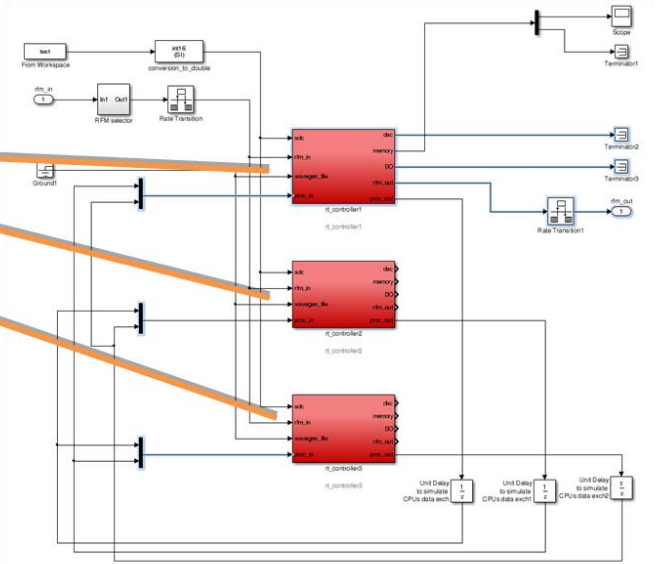
Legacy TCV control systems backplane
- Supports all already developed control electronics systems (since 30 years)
- Allows integration of new and modern control communication networks into legacy control hardware

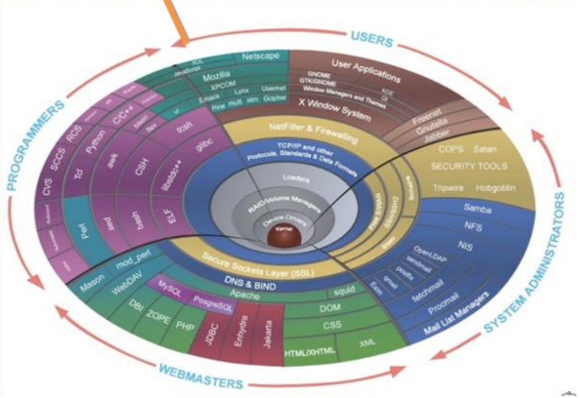
Photo courtesy of Swiss Plasma Center Electronics Lab

Central processing units (CPUs)



Intel I7 architecture

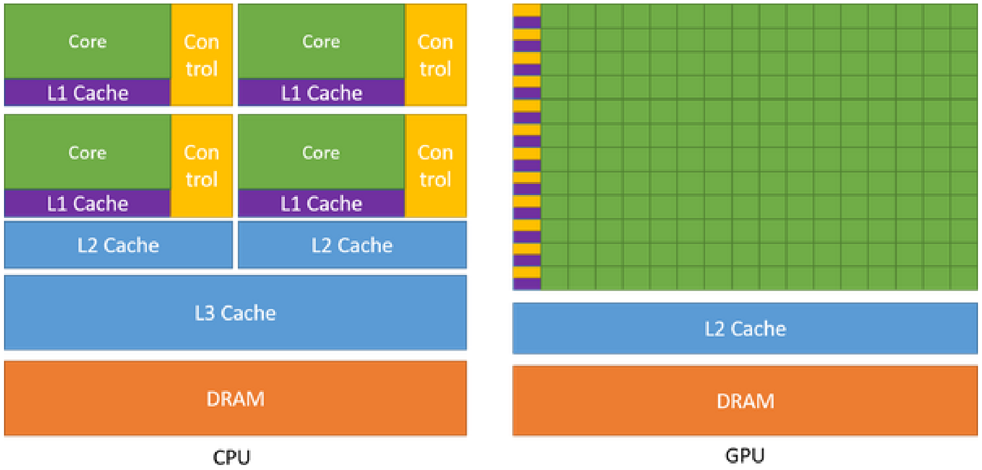




All complex control systems nowadays run on multi-core CPU architectures, a lot of Linux based RT computer tend to put the kernel on the first core, leaving the other free for RT computation (but highly s.o. dependent)

Swiss Plasma Center

Graphics processing units (GPUs)




```
// Kernel definition
__global__ void VecAdd(float* A, float* B, float* C)
{
    int i = threadIdx.x;
    C[i] = A[i] + B[i];
}

int main()
{
    ...
    // Kernel invocation with N threads
    VecAdd<<<1, N>>>(A, B, C);
    ...
}
```

- GPU offer a large number of parallel execution threads, hence are the primary candidate to execute machine learning related controls (i.e. neural networks) or other computationally intensive parallelizable data processing algorithms
- Not used so far on TCV (NN executed on conventional CPUs, but advances in interface software will facilitate their integration)

https://docs.nvidia.com/cuda/cuda-c-programming-guide/
https://ch.mathworks.com/help/signal/gpu-code-generation.html

MathWorks® R2024b

GPU Code Generation

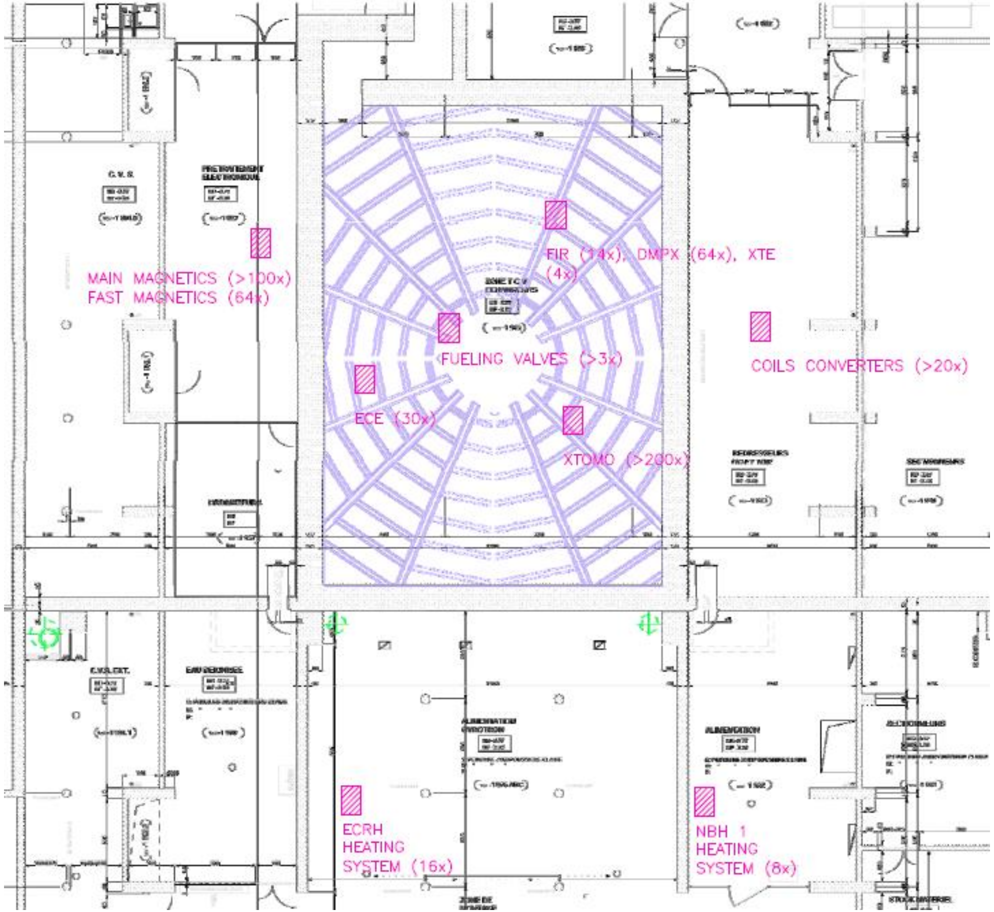
Generate CUDA® code from MATLAB® code

Real-time networks

EPFL

TCV's control system layout, demand of real-time networks

- Tokamaks are spatially distributed experiments, hence they require inherently distributed control systems
- For data distribution among systems -> need for realtime data networks
- For time distribution among systems -> need for distributed timing systems

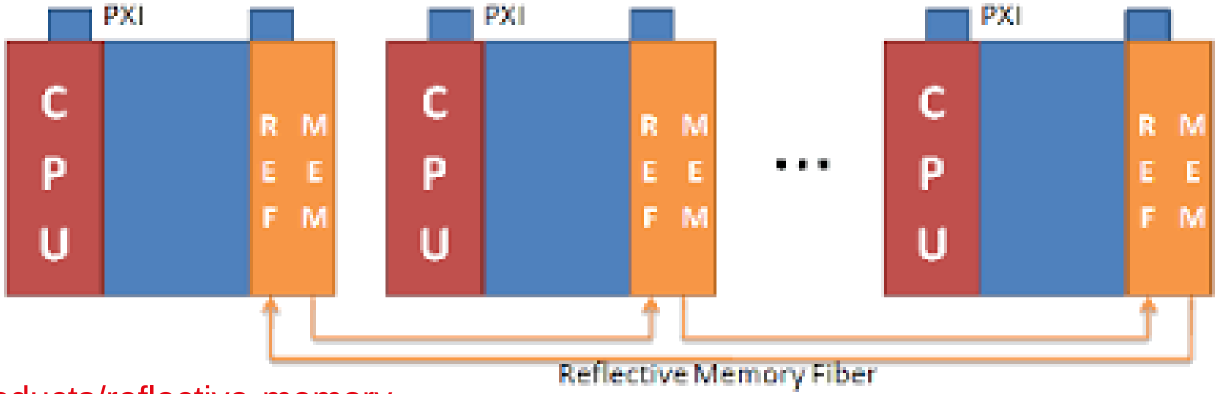


Swiss Plasma Center

Reflective memory networks

Galperti Cristian

Serial communication channels have to guarantee the minimum latency time to transfer data from one point to the next. RFM is a hardware/software agnostic fast (>2gbps) fiber optic link which automatically synchronizes memories between several hosts.

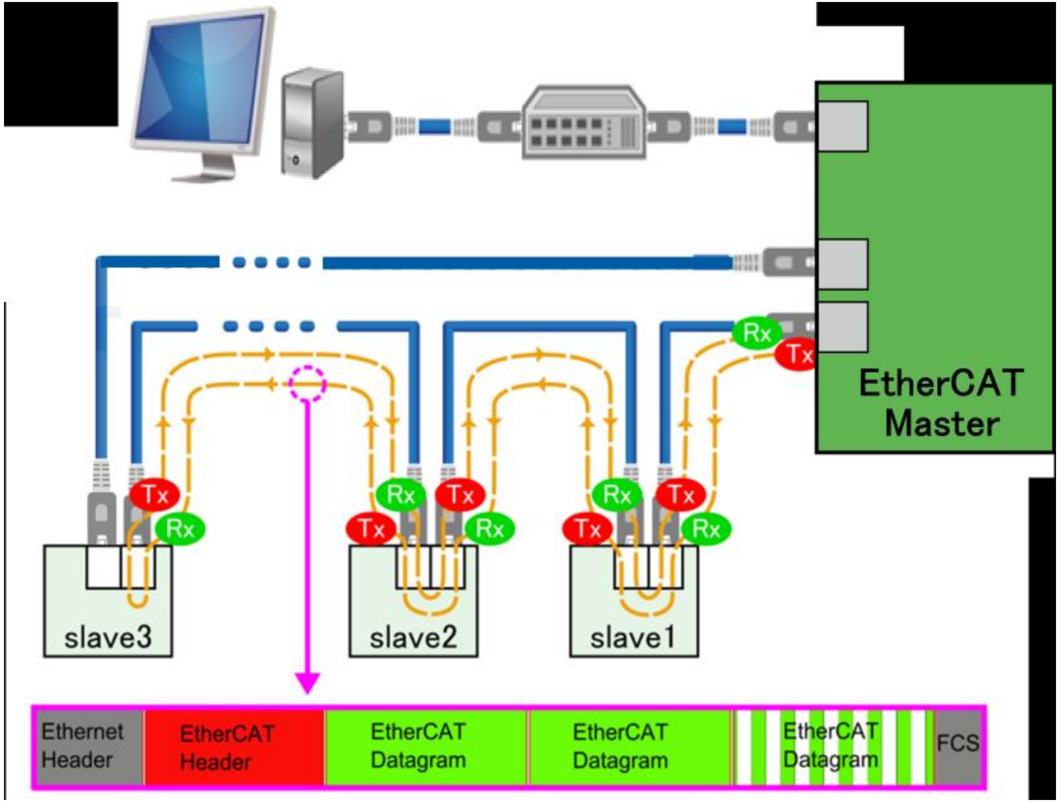


Swiss Plasma Center

https://www.abaco.com/products/reflective-memory

EPFL

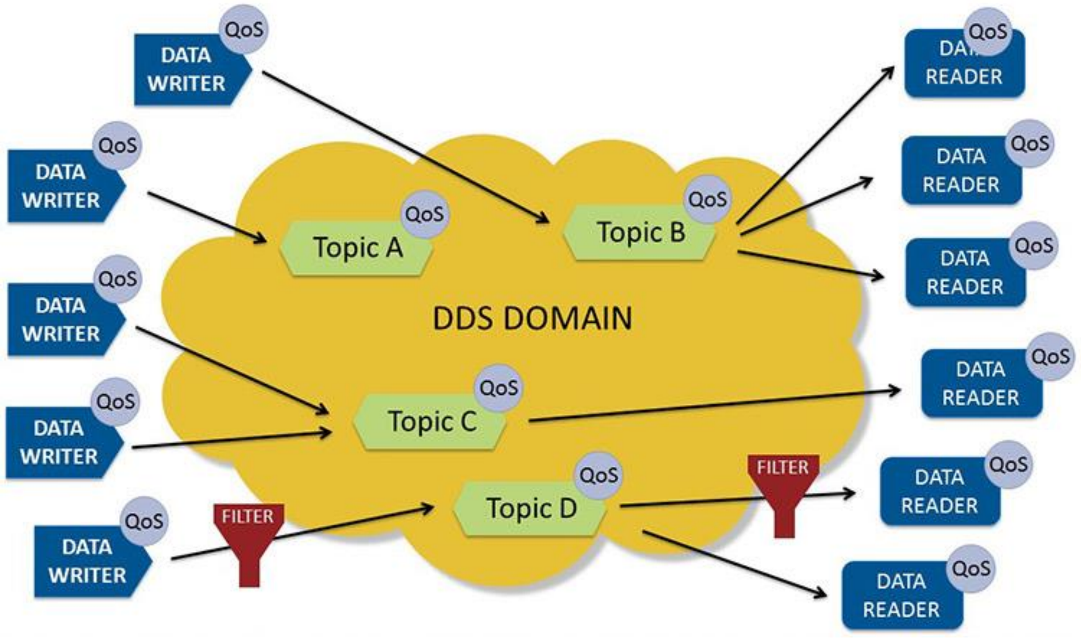
EtherCAT networks



EtherCAT (Ethernet for Automation and Control Technology) is an Ethernet based control network. The goal during development of EtherCAT was to apply Ethernet for automation applications requiring short data update times (also called cycle times; ≤ 100 μs) with low communication jitter (for precise synchronization purposes; ≤ 1 μs) and reduced hardware costs. (https://en.wikipedia.org/wiki/EtherCAT)

Galperti Cristian

DDS networks (publish / subscribe concept)

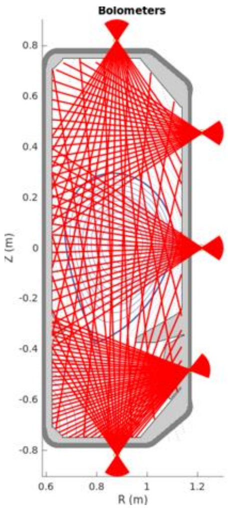


DDS (Data Distribution Service) is a middleware protocol and API standard for data-centric connectivity.

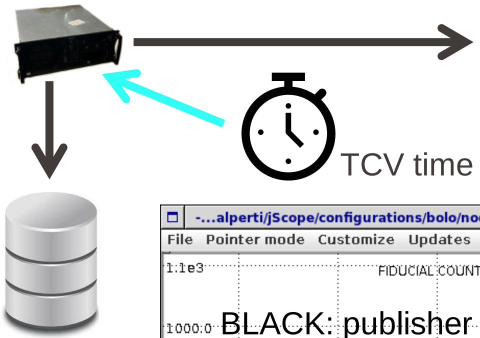
- Data is published by data-writers and subscribed to by data readers.
- Highly scalable, only data dependent.

www.dds-foundation.org

DDS network TCV example

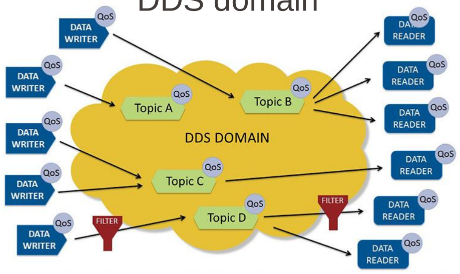


Bolometry "topic" publisher
- TCV synchronous
- 144 lines of sight, 10 kHz sample rate
- 1 kHz publication rate
- Fiducial counter part of the topic

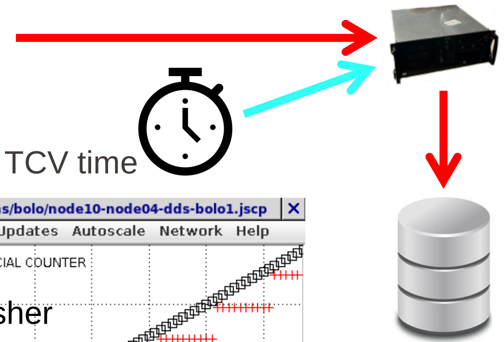


MDSplus

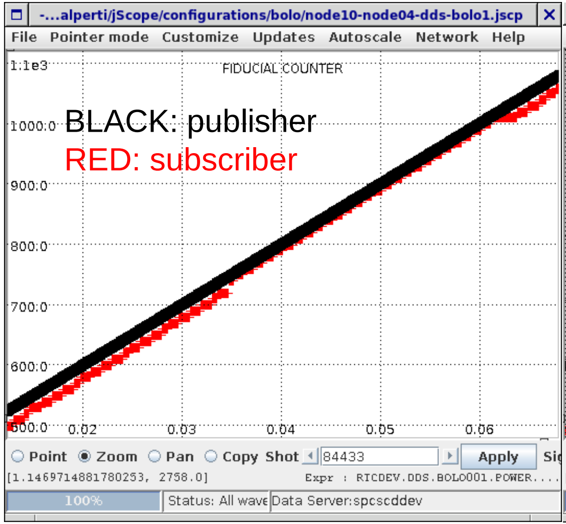
DDS domain

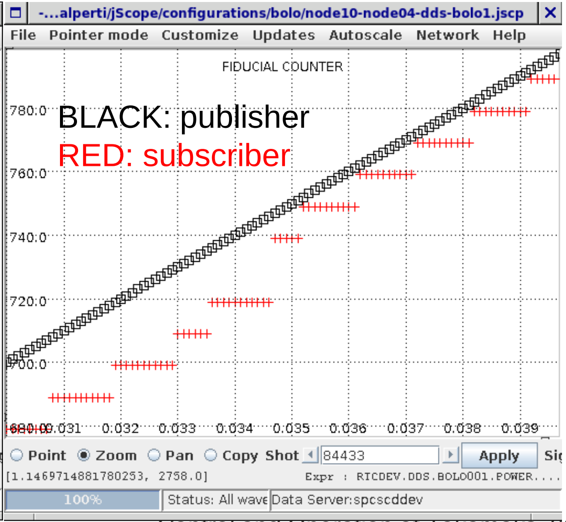


Bolometry "topic" subscriber
- TCV synchronous
- Asynchronous (non blocking) subscription

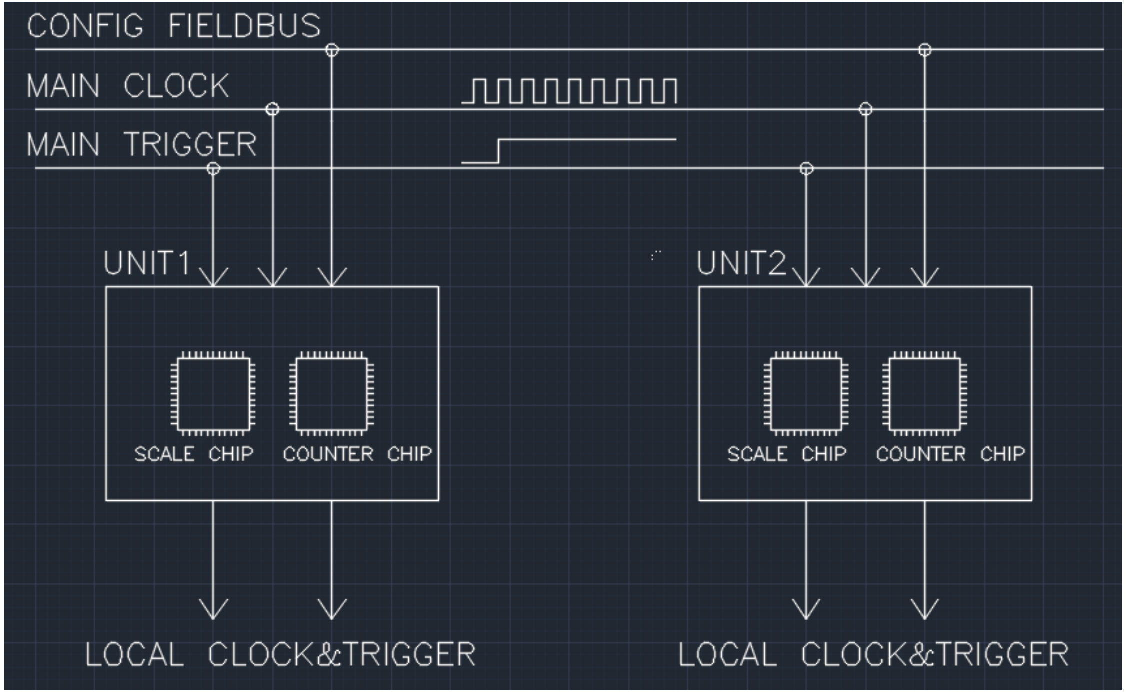


MDSplus





Distributed timing systems, hardwired

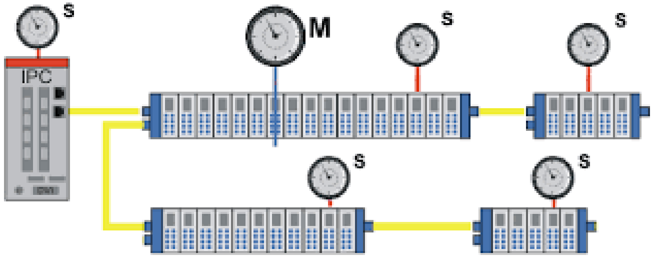


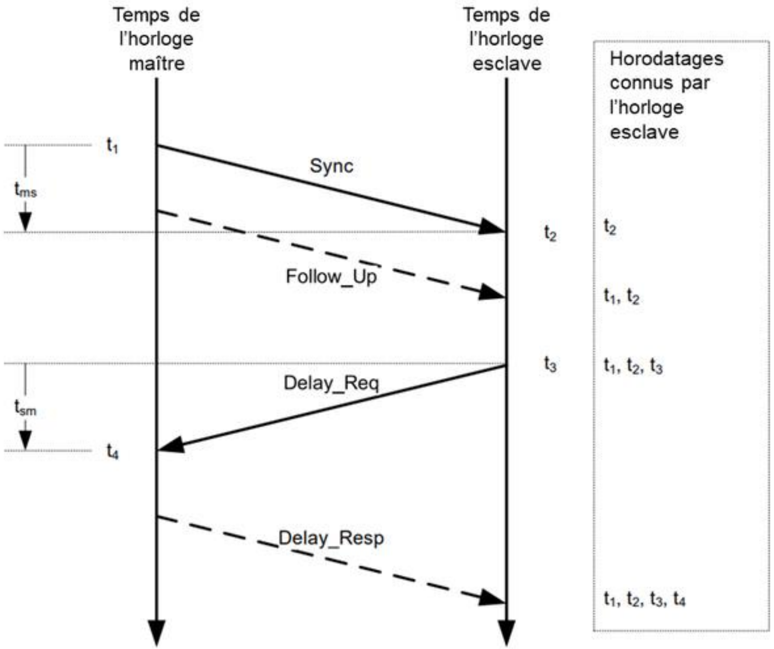
- First sync concept deployed on tokamaks
- Pros:
  - architecturally simple.
  - maintainable and upgradeable for a long time
- Cons:
  - Best sync accuracy: 1 us
  - No 24h/24 sync (only pulsed sync)
  - No sync groups
  - No industrial support

Swiss Plasma Center

Distributed timing systems, network based

Galperti Cristian



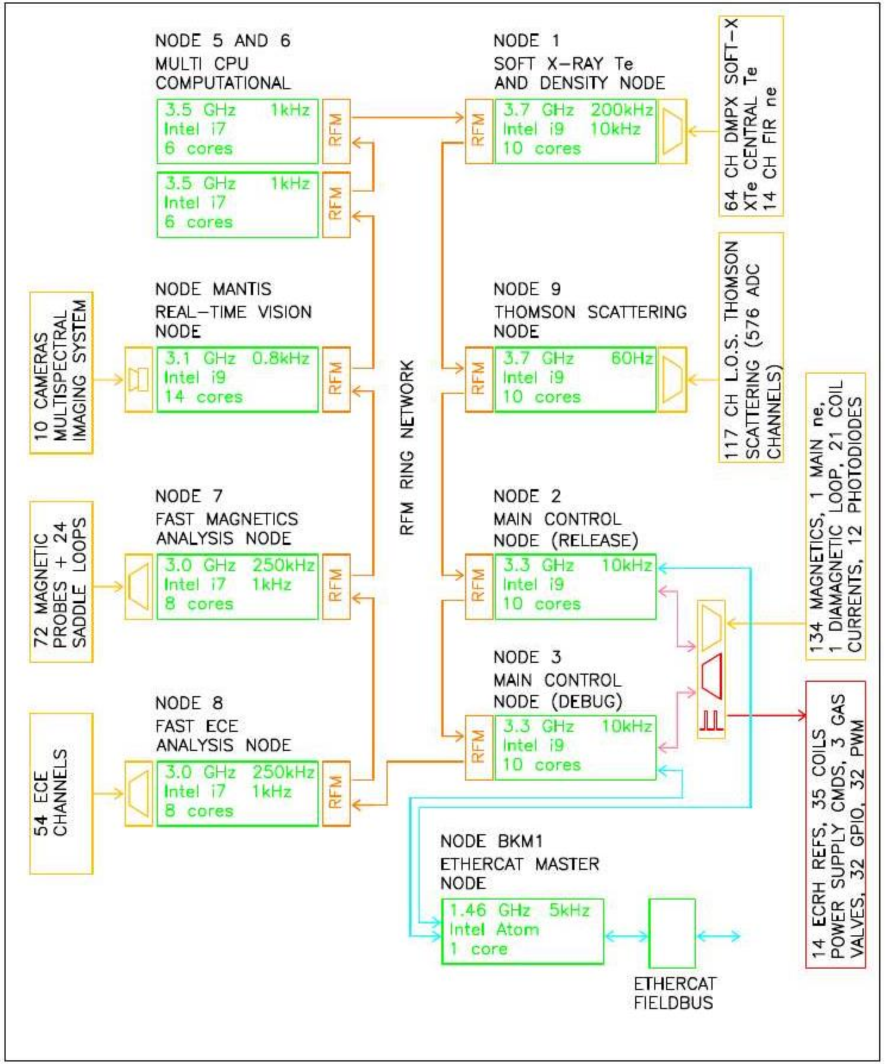


- Current industry & research standard
- Pros:
  - Best sync accuracy: 1 ns (research, WhiteRabbit project), 100 ns (industry, PTP and EtherCAT DC).
  - 24h/24 sync and sync groups out of the box
  - Strong industrial and research support
  - Easy deployable and debuggable (it is like a computer network)
- Cons:
  - Still low widespread availability of control equipment based on them, but situation is evolving rapidly.
- http://white-rabbit.web.cern.ch/
- https://en.wikipedia.org/wiki/Precision_Time_Protocol

TCV's digital control system hardware

Swiss Plasma Center

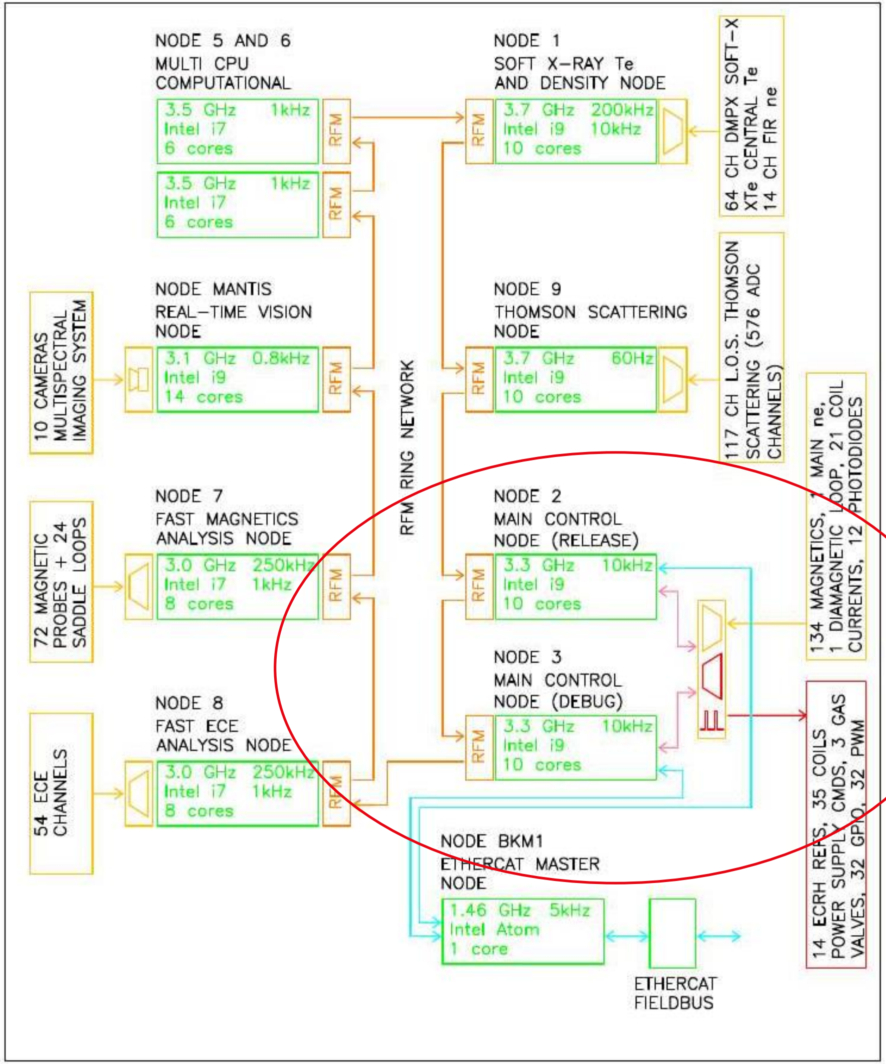
TCV digital distributed control system (SCD)



- Various types of interconnected control subsystems:
  - Low latency systems for hard real time control at fastest rate
  - Oversampling systems for fast diagnostics acquisition followed by complex real-time analysis algorithms
  - Multi-core computational systems for CPU hungry codes
  - Real-time vision nodes for vision in the loop systems
  - Not isochronous system (real-time thomson scattering)

Galperti Cristian

TCV digital distributed control system (SCD)

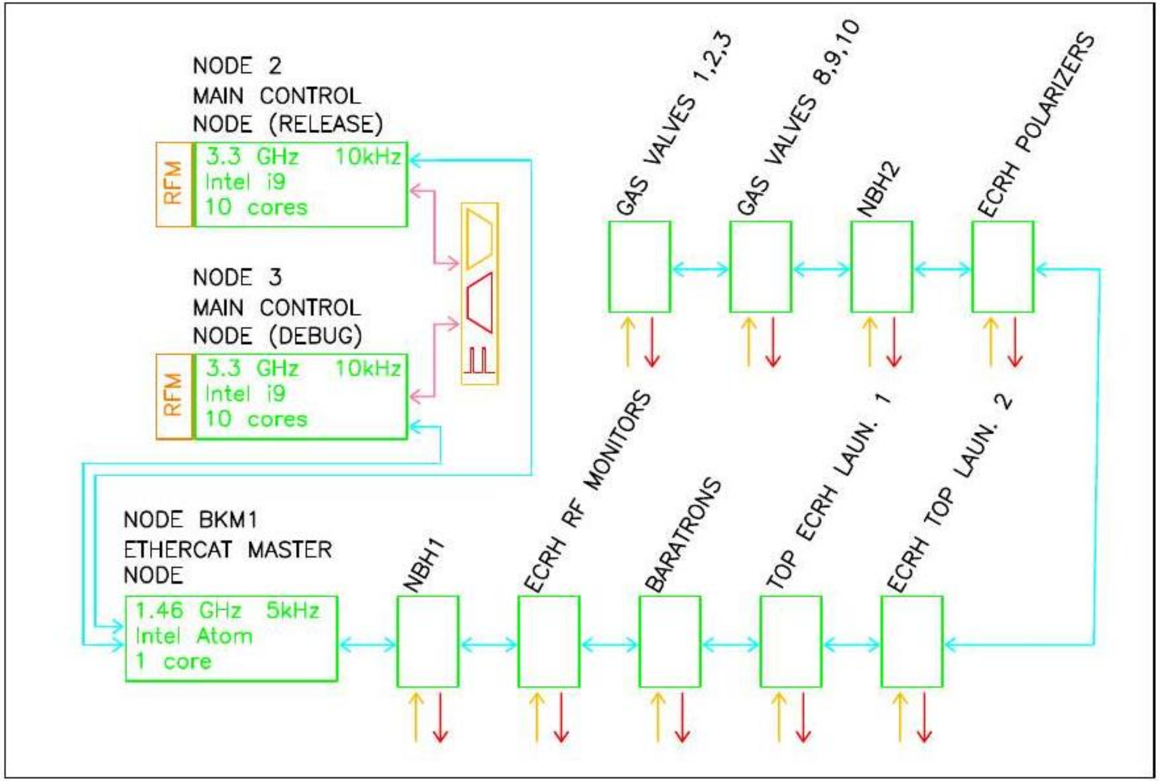


- Central main plasma control system duplicated on two identical machines
- Reflective memory as real-time network backbone
- Multi-core CPU only based (no GPUs to date)
- EtherCAT for fast, fexible, distributed and cost effective I/O interconnection

https://doi.org/10.1016/j.fusengdes.2024.114640

TCV digital distributed control system, EtherCAT

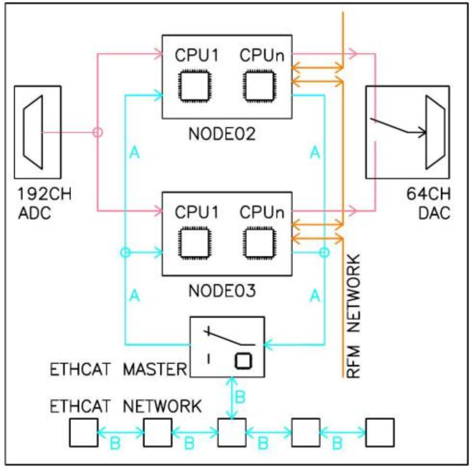
Galperti Cristian



- Central main plasma control system is interfaced with a realtime EtherCAT network for flexible I/O interconnections, and direct digital drive of actuators

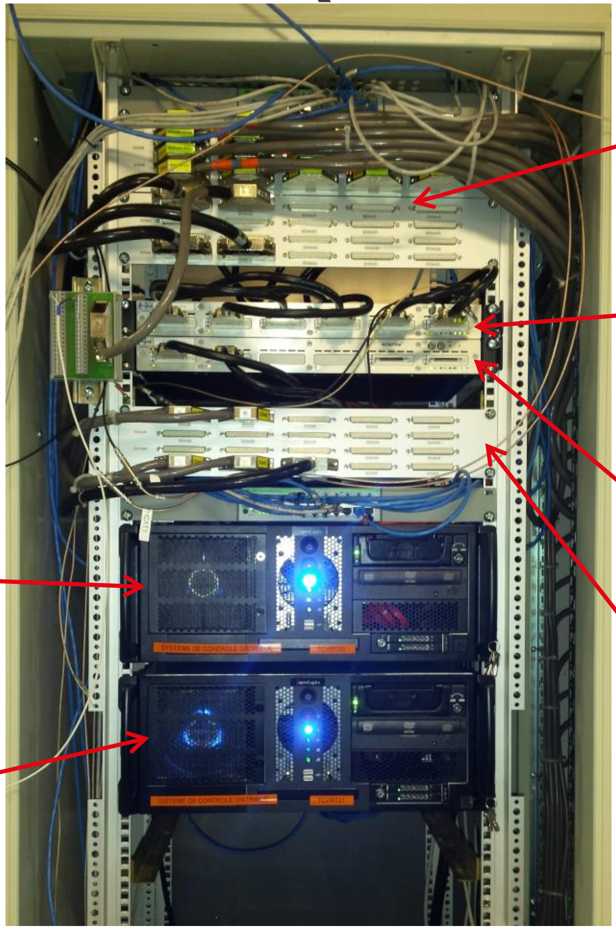
TCV main control nodes (node 02 and 03)

Galperti Cristian



Node 02
Industrial PC
(tcvrt20.crpp.tcv)

Node 03
Industrial PC
(tcvrt21.crpp.tcv)



Inputs patch panels

ADC input system (acq2106_076
192 channels up to 1 MHz)

DAC + PWM output system (acq2106_079, 64 (up to 128) channels and 32 PWMs)

Outputs patch panels

Actuators

Tokamak actuators, a list

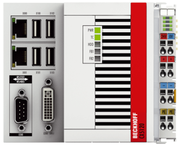
- Coils voltage power supply. They are usually multi MW controllable voltage or current mode power converters either using thrystors or high power transistors technologies.
- Gas injection systems, either composed of flux controllable gas valves or pellet injectors.
- Auxiliary RF based heating actuators, usually MW grade RF generators with associated power supplies. They usually encompass mechatronics or radio-frequency components to steer the RF beams.
- Auxiliary neutral beams based heating actuators, usually composed of an ionized gas generator, a linear accelerator and a gas neutralizer.

Galperti Cristian

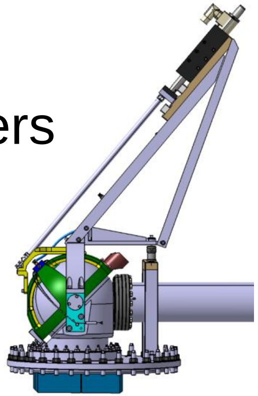
Direct digital drive of actuators, examples

- Linux EtherCAT master stack
- All digital realtime I/O with actuators
  - ECRH_L10+L11 mirror angle command
  - ECRH_L10+L11 mirror angle measure
  - All machine coils voltage command
  - All machine coils voltage and current measures
- https://esd.eu/en/products/ethercat-master

ECRH waves top launchers

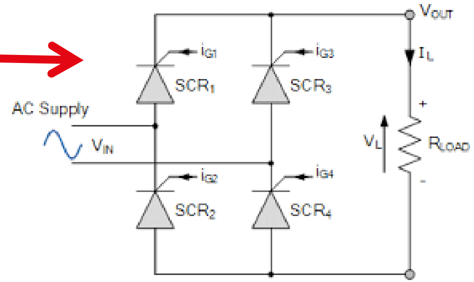


https://www.beckhoff.com/en-en/



Main coils power supplies controller

https://imperix.com

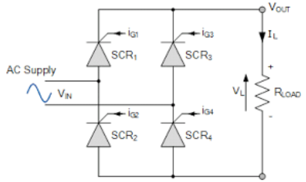


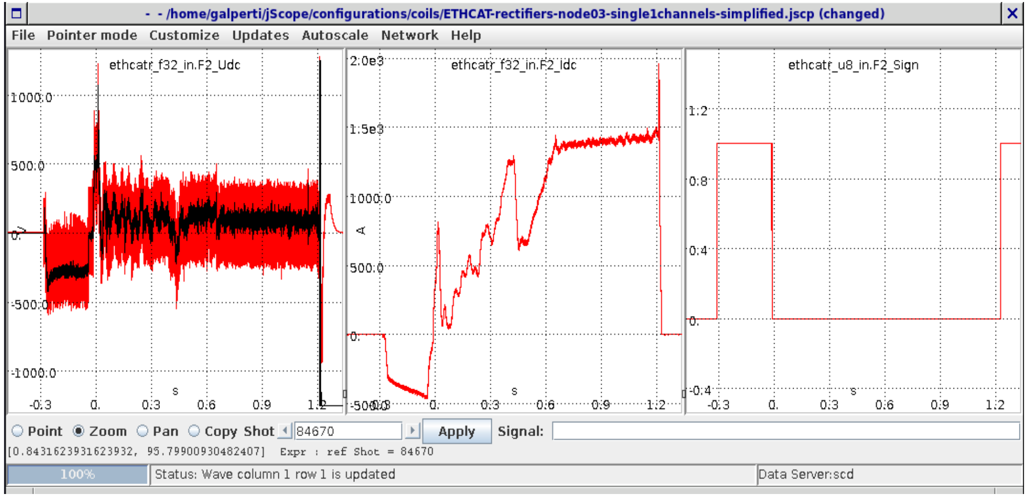
Direct digital drive of actuators, F coil example

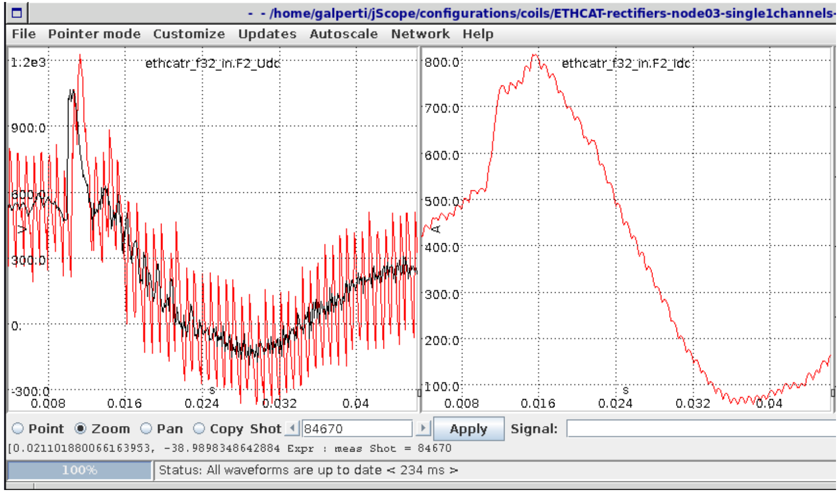
Magnetic controller

114 signals, 10kHz refresh rate on realtime network

Controller + SCR rectifier

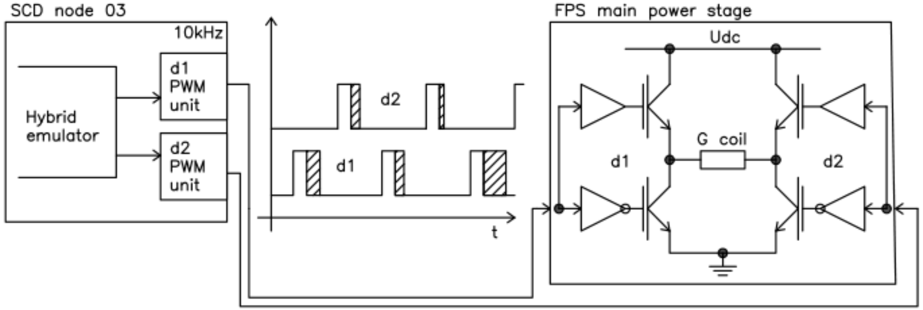


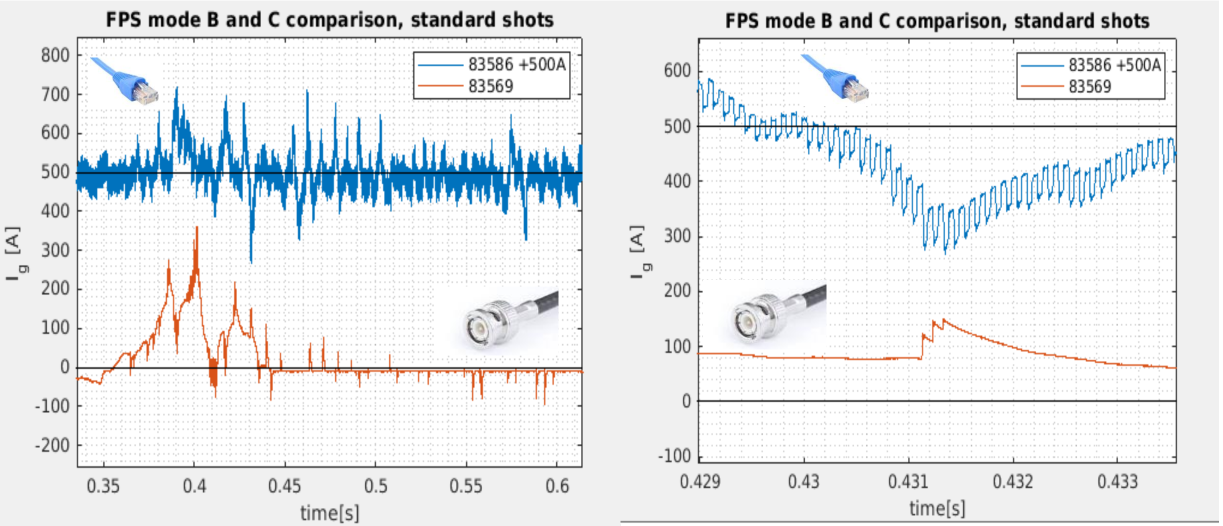




Black F2 voltage ref, RED F2 converter measures

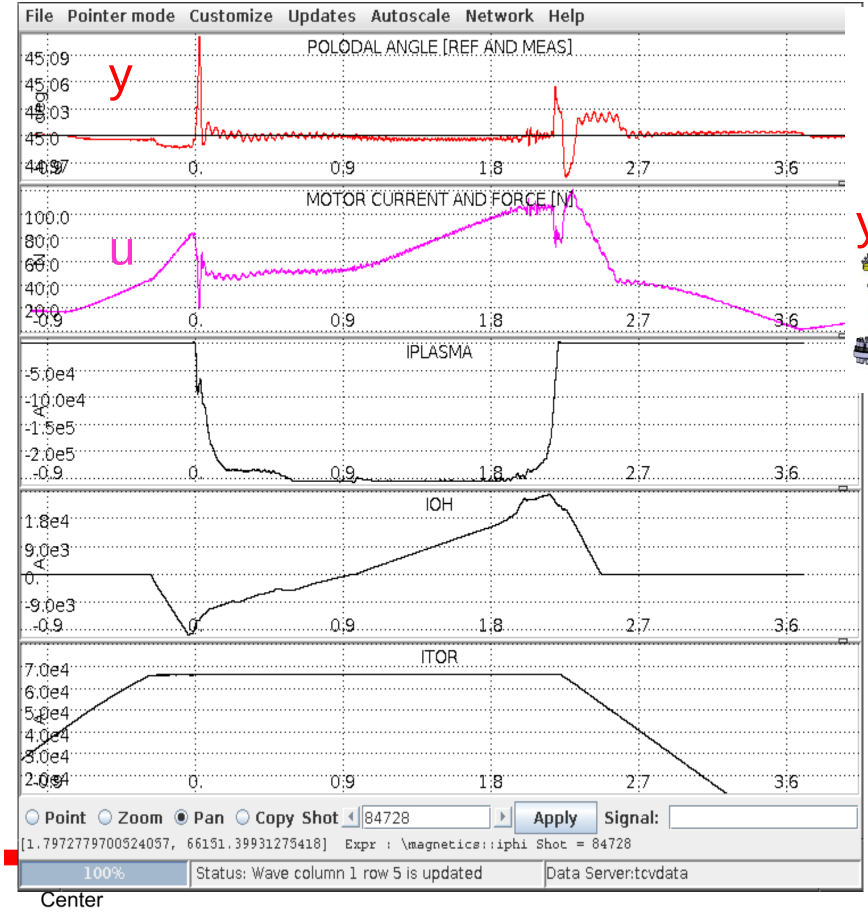
Direct digital drive of actuators, G coil example

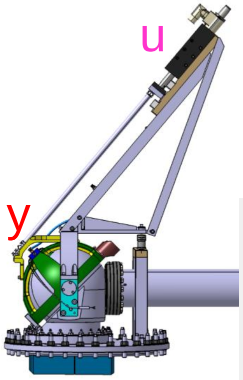




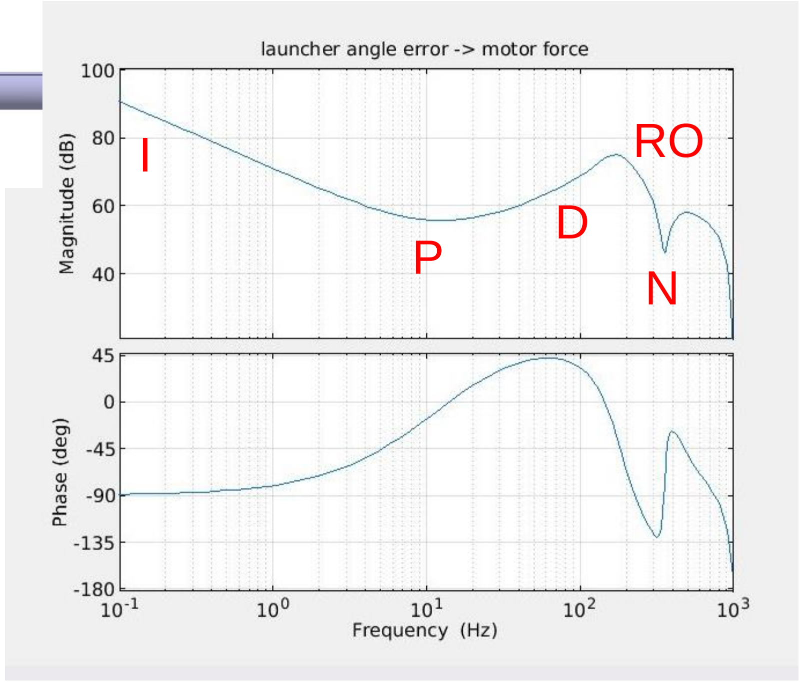
- TCV internal vertical stabilization coil power supply can be driven digitally, with direct access to the two sides its H bridge
- SCD is equipped with two independent phase and duty adjustable PWM modulators to do this
- G actuation linearity around Ig=0 greatly improves

SISO actuator example, disturbances rejection



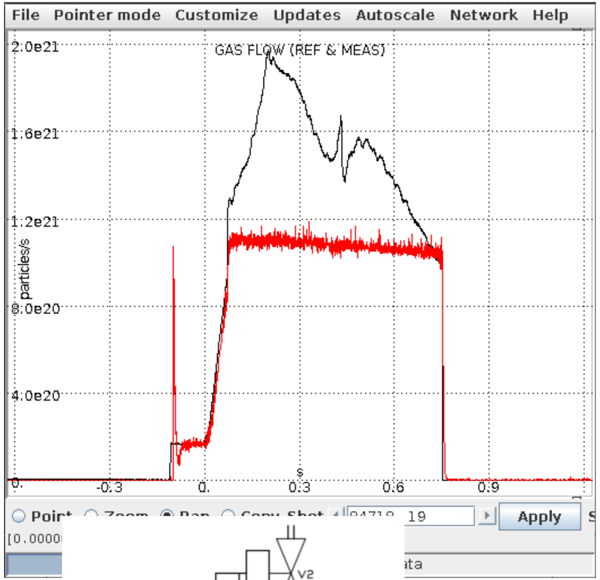


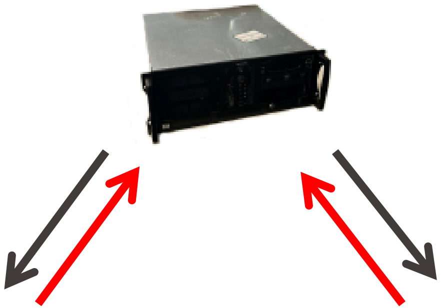
- PID with D roll off and notch filter on a servo-actuator resonance
- High low frequency gain important for disturbances rejection

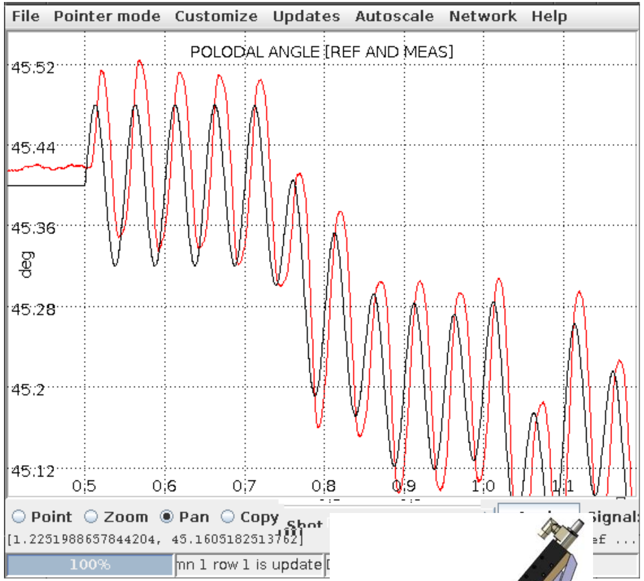


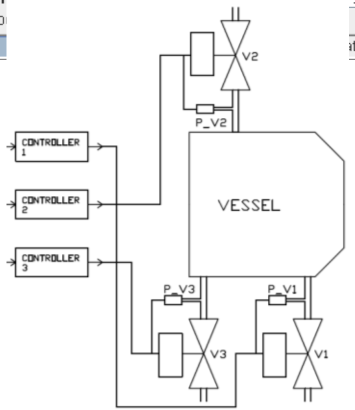
Galperti Cristian

Other actuators non-idealities examples



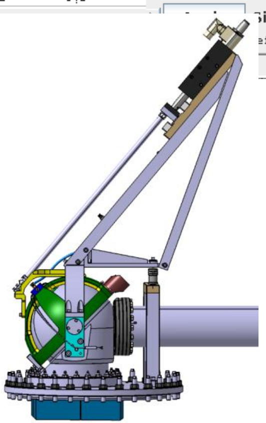






- Main D2 fueling gas valve flow saturation

- ECRH launch angle phase shift during extremum seeking tracking

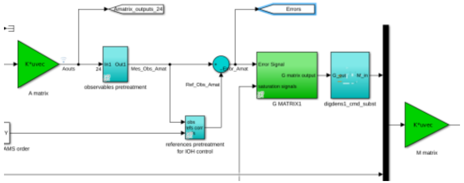


Swiss Plasma Center

Tokamak control systems software aspects

Tokamak control systems, three key elements

Control system development
- Controllers design
- System simulation, verification & validation



SIM – EXEC path

Real time control system
- Control execution

SIM – DB path

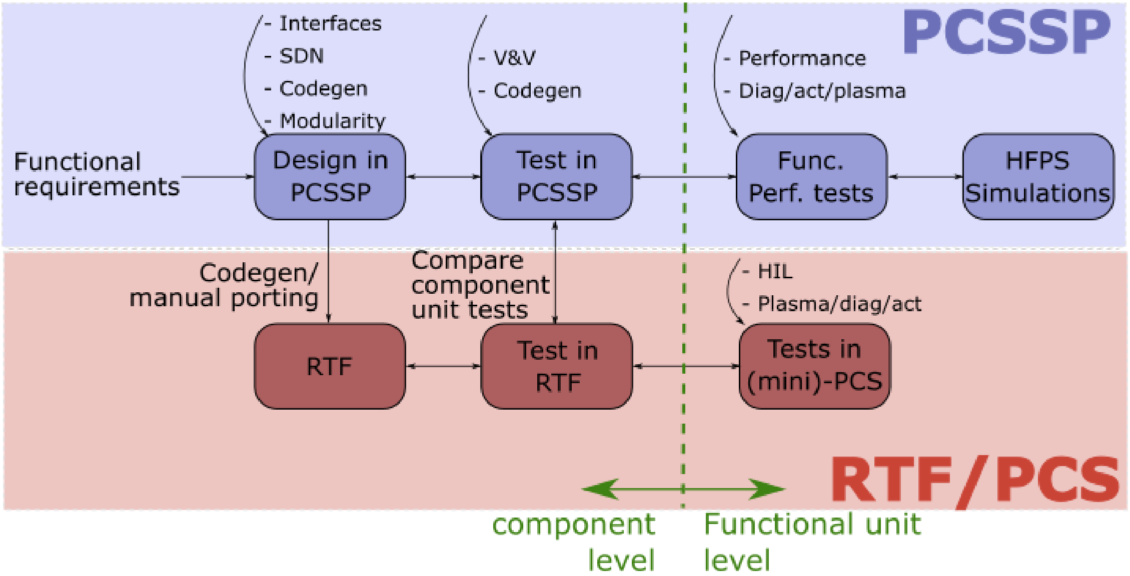
EXEC – DB path

Experimental databases
- Previous experiments data and parametrization
- Test results
- Next experiment parametrization, experiment data archiving

Galperti Cristian

Swiss Plasma Center

ITER example



Control system simulator
- Control design and V&V

SIM – EXEC path

Real time control system
- Control execution

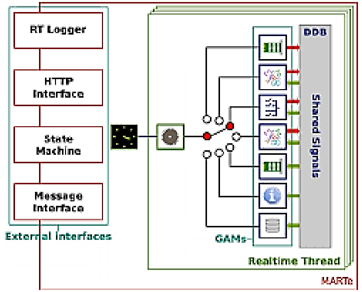
Swiss Plasma Center

TCV realtime digital control system case

Galperti Cristian

SIM – EXEC path

MATLAB/Simulink + SCDDS framework (Control code development and simulation)



MARTe2 (Real time control framework)

SIM – DB path

EXEC – DB path

MDSplus (shot parametrization and data archiver)

https://doi.org/10.1016/j.fusengdes.2024.114640

EPFL

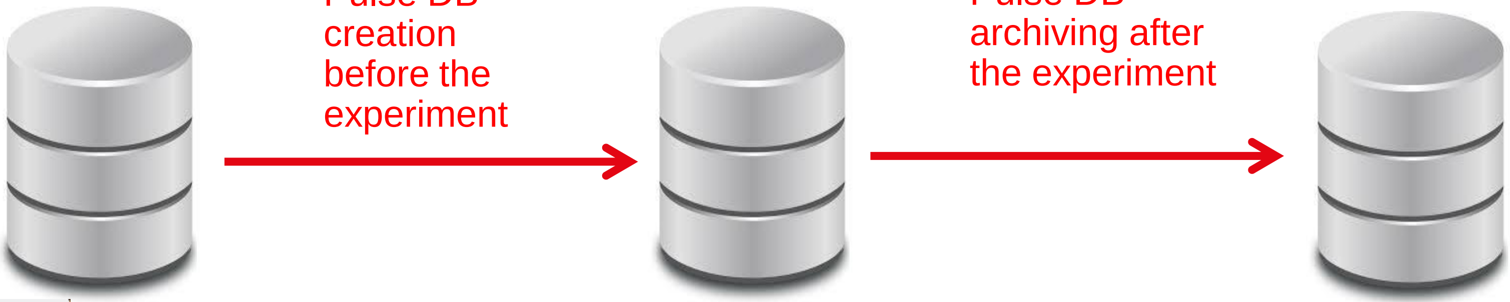
MDSplus data archiving system

Galperti Cristian

Model DB

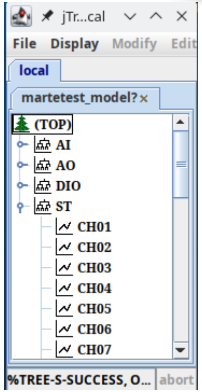
Live pulse DB

Archived pulse DB



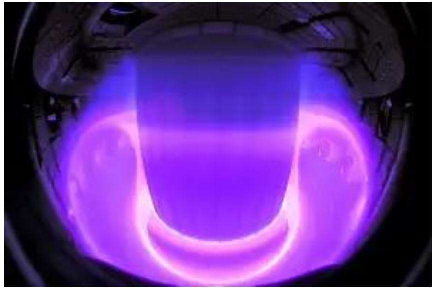
Pulse DB creation before the experiment

Pulse DB archiving after the experiment

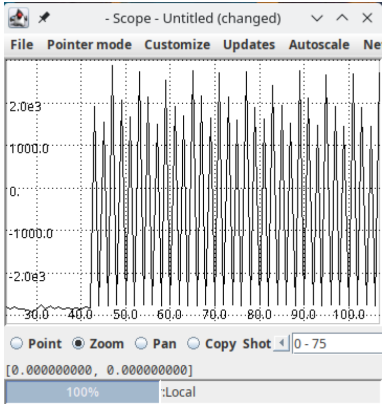


Pulse config

Pulse execution



Offline analysis



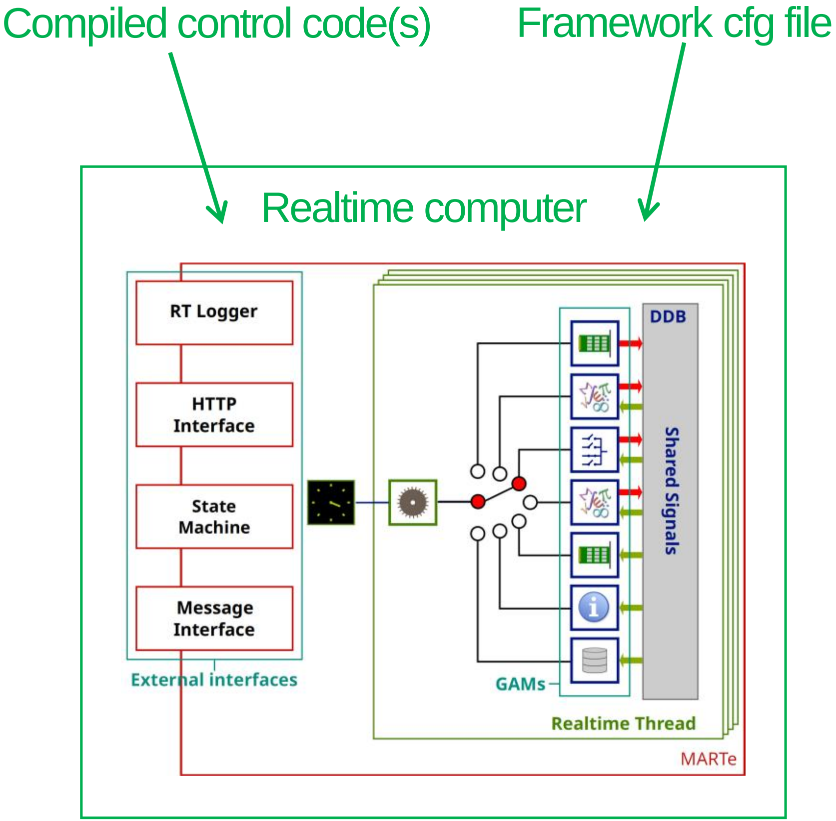
Swiss Plasma Center

www.mdsplus.org

MARTe2 framework

MARTe2 is a C++ software framework conceived to help building and running real-time applications. TCV almost entirely switched to it from 2019.

- Standardization and modularization of realtime control applications
- Multithread/multicore applications natively supported
- Component to load Simulink generated code, with introspection
- Components to read/write MDSplus entries
- State machine / messages interface, now via OPC/UA server
- Extensive logging



https://vcis.f4e.europa.eu/marte2-docs/master/html/

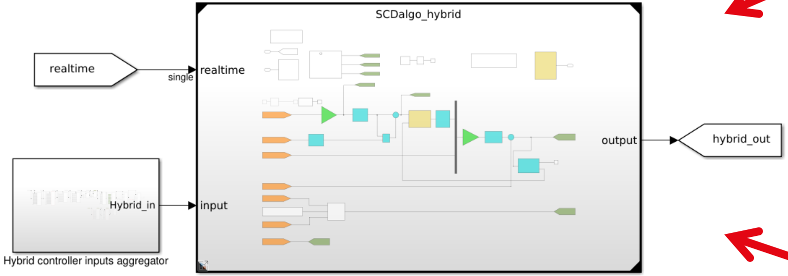
TCV's digital control system software, applications

Simulink (and its paths)

Galperti Cristian

MATLAB/Simulink®

Simulink modeled control system part (e.g. algorithm)

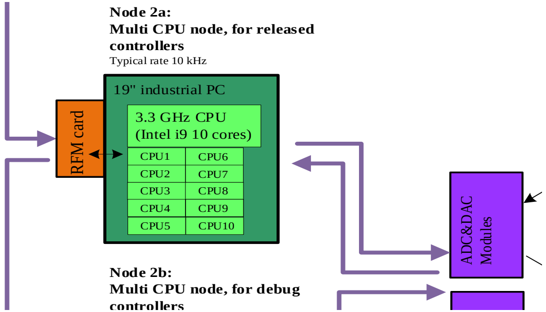


SIM – EXEC path

SIM – DB path

Real-time control systems

PLANT



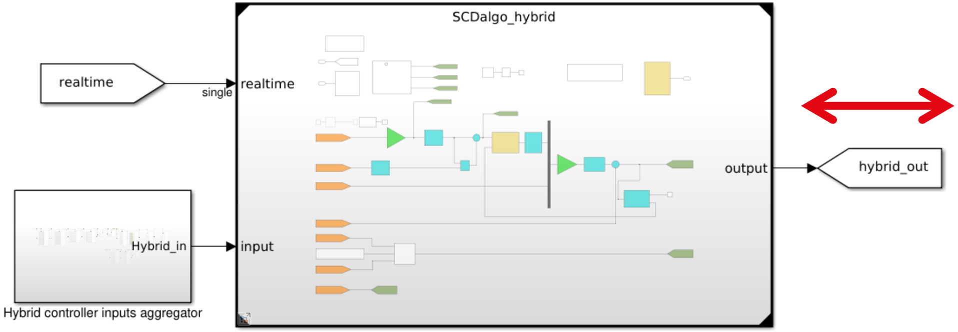
Database systems

Swiss Plasma Center

The SCDDS framework

Simulink Control Development and Deployment Suite

Simulink modeled control system part (e.g. algorithm)



SCDDS framework, a MATLAB object oriented framework to handle and interface control code

```
classdef (Abstract) SCDDSclass_algo
    %SCD algorithm handling object
    %   The class holds all information and
    %   methods for handling a Simulink
    %   algorithm

    properties (SetAccess = private, Hidden=false)
        modelname              % Name of the model
        modelslx               % slx model file name
        folder                 % folder containing algorithm
        datadictionary         % Name of the used data dictionary
    end
```

```
%%
classdef SCD_algo < SCDDSclass_algo

end
```

```
function [obj] = SCDalgoobj_hybrid()

%% Hybrid controller core algorithm
obj=SCD_algo('SCDalgo_hybrid');

%% Timing of the algorithm
obj=obj.settiming(-4.5,1e-4,3);

%% Tunable parameters structure name
obj=obj.addtunparamstruct('SCDalgo_hybrid_tp');

%% Tunable parameters
obj=obj.addparameter(SCDclass_mdsparmatrix
obj=obj.addparameter(SCDclass_mdsparmatrix
```

Swiss Plasma Center

gitlab.epfl.ch/spc/scdds
LGPL open source license

perti Cristian

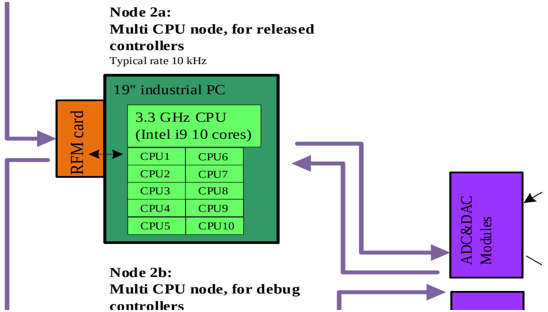
SCDDS framework, overview

Set of instantiated objects in MATLAB representing the control system

```
function [obj] = SCDalgoobj_hybrid()

%% Hybrid controller core algorithm
obj=SCD_algo('SCDalgo_hybrid');

%% Timing of the algorithm
obj=obj.settiming(-4.5,1e-4,3);

%% Tunable parameters structure name
obj=obj.addtunparamstruct('SCDalgo_hybrid_tp');

%% Tunable parameters
obj=obj.addparameter(SCDclass_mdsparmatrix
obj=obj.addparameter(SCDclass_mdsparmatrix
```

- Code generation and deployment
- MARTe2 cfg files generation and deployment

Real-time control systems (MARTe2 based)

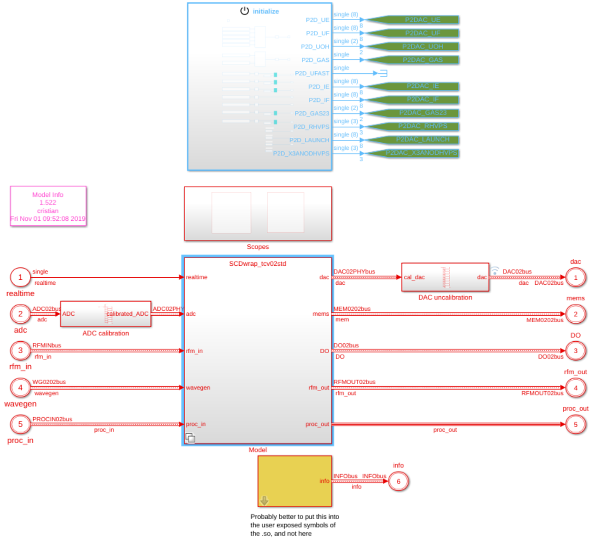


- Parameters and waveforms retrieval for code simulation
- Parameters deployment for shot execution

Database systems (MDSplus based)

SCDDS, code generation from Simulink

Simulink algorithm with tunable parameters



Simulink coder with enabled CAPI + C compiler (gcc or icc)

Linux shared library

Introspective interface for I/O ports

Introspective interface for tunable parameters

Introspective interface for internal states

Introspective interface for internal signals

Galperti Cristian

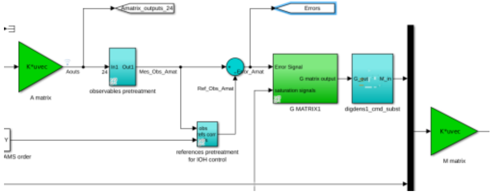
SCDDS simulation and MARTe2 execution workflow

Galperti Cristian

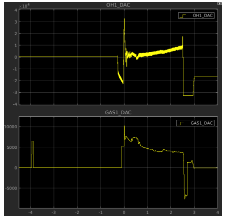
MDS+ parameters and waveforms DB

SCDDS (MATLAB) parameters and waveforms loaders

Simulink model under SCDDS framework



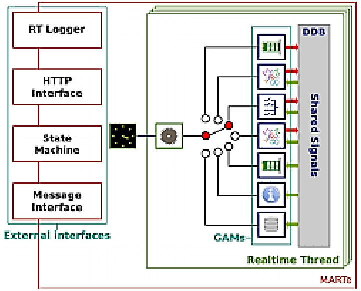
Simulink simulation



Automatic code generation

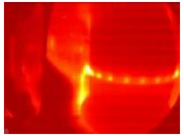
=

MARTe2 (C++) parameters and waveforms loaders



MARTe2 control system

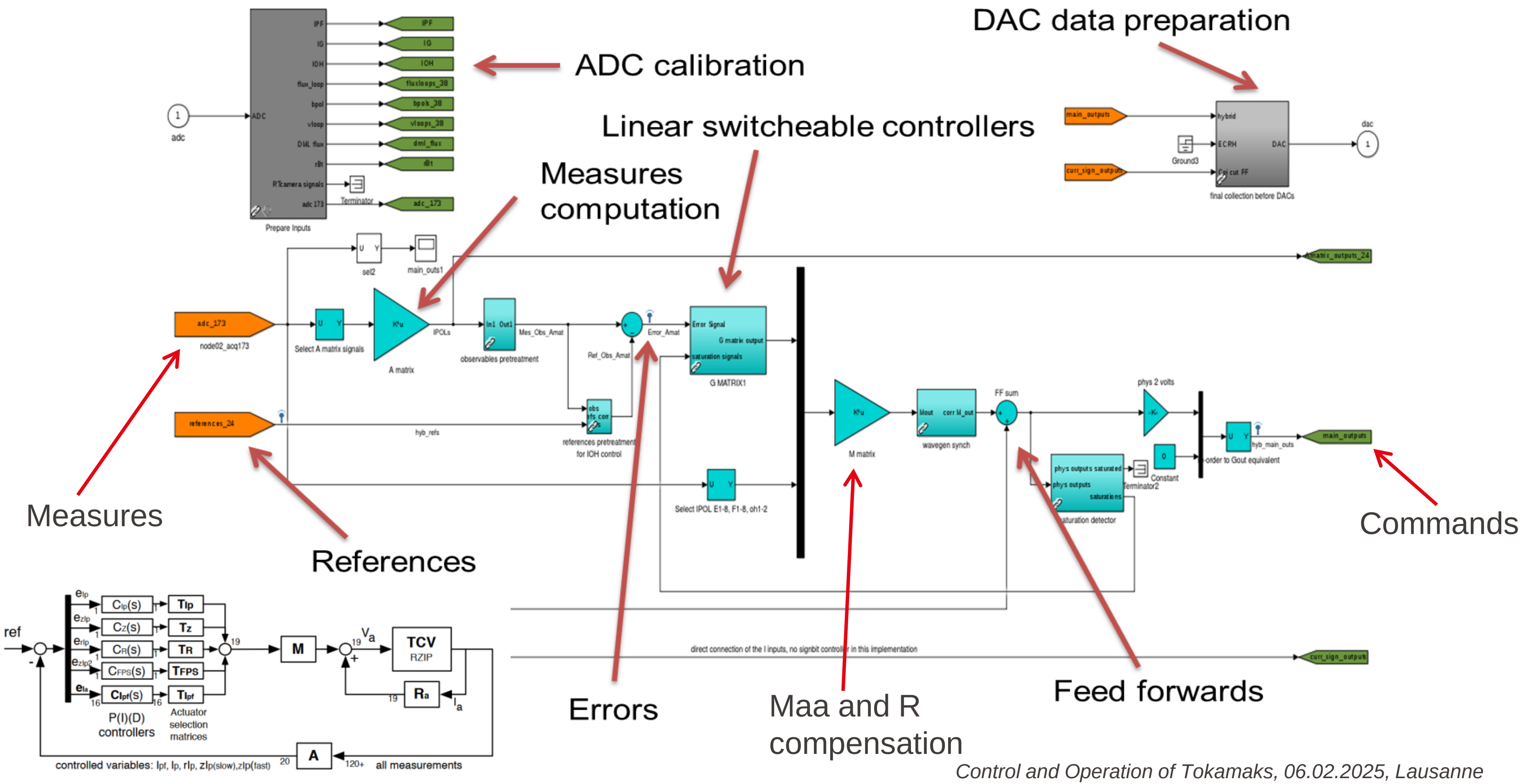
RT execution



MDS+ shot data

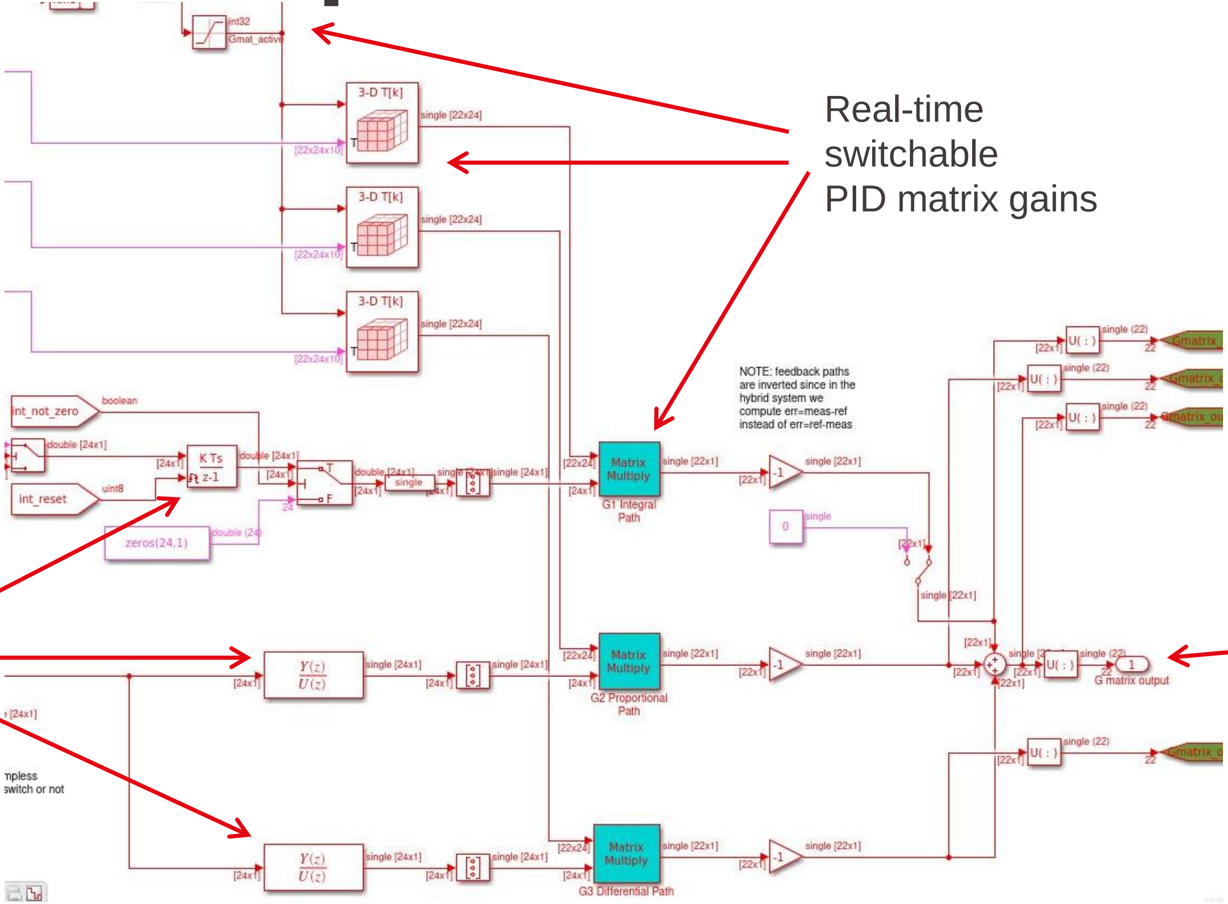
TCV standard plasma control code

Galperti Cristian



TCV standard plasma control code

Real-time switchable PID matrix gains



24 channels PID Transfer functions

PID Commands

TCV Simulink plasma control simulation example

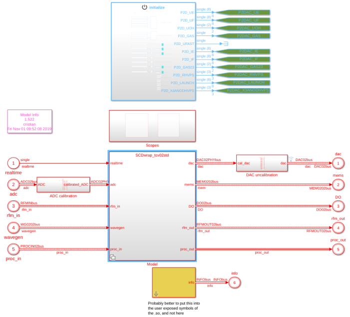
Galperti Cristian

SCDDS
loader

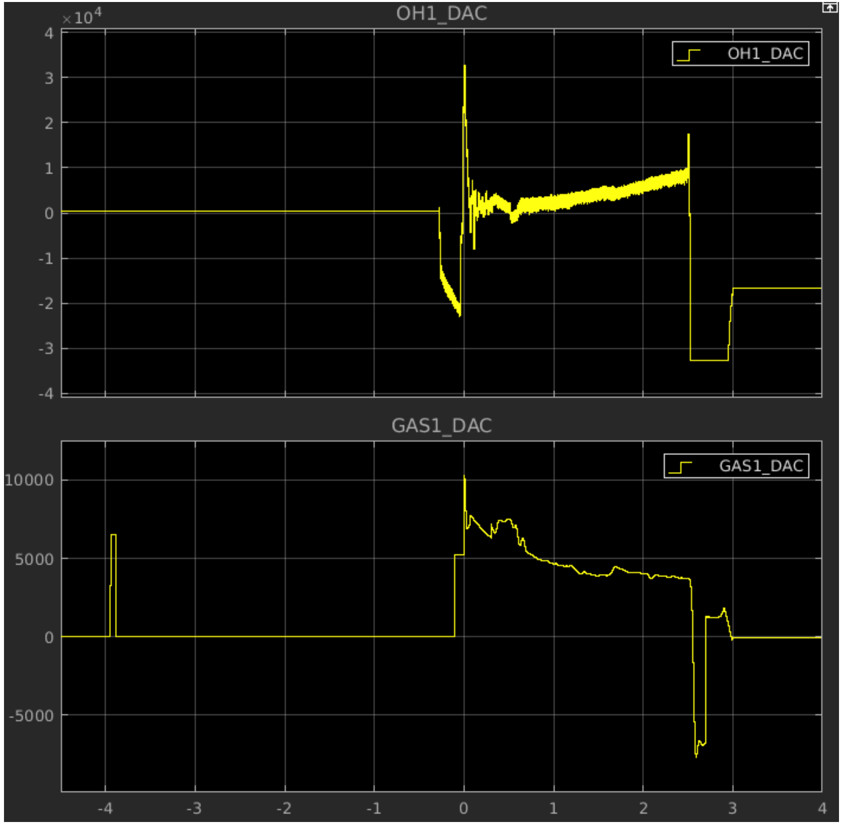
Shot parameters
From MDS+

Shot ADC data
From MDS+

SCDDS
loader



Shot simulation



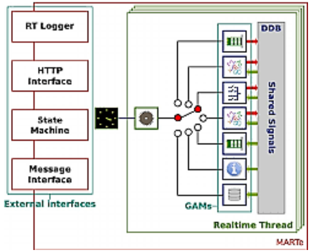
Swiss
Plasma
Center

TCV real-time shot reprocessing with MARTe2 example

Galperti Cristian

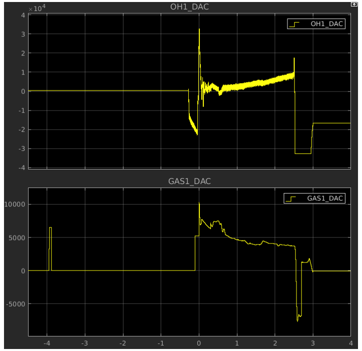
MARTe2 loader

Shot parameters
From MDS+

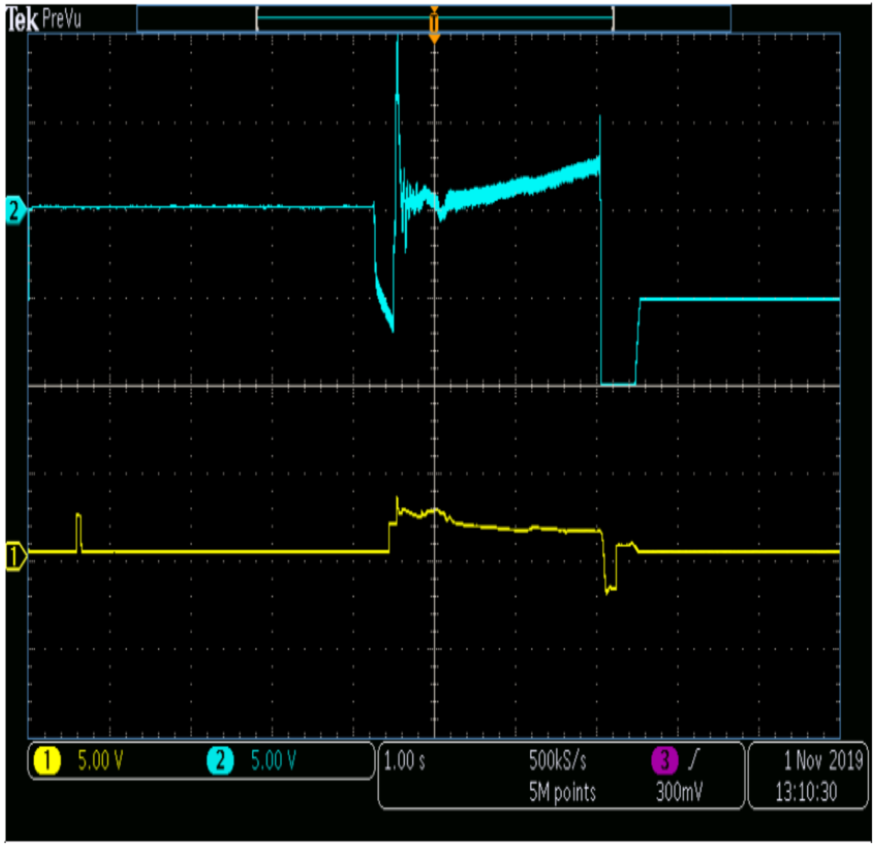


Shot ADC data
From MDS+

MARTe2 loader



RT reprocess run



Swiss Plasma Center

EPFL

Multi CPU control layout on TCV

Galperti Cristian

CPU#2

- Main linear magnetic controller
- Fast diagnostics and actuator interfaces

CPU#3

- MEQ solver (RTLIUQE)
- Vertical growth rate estimator
- Disruption proximity estimator
- Active plasma shape controller

CPU#4

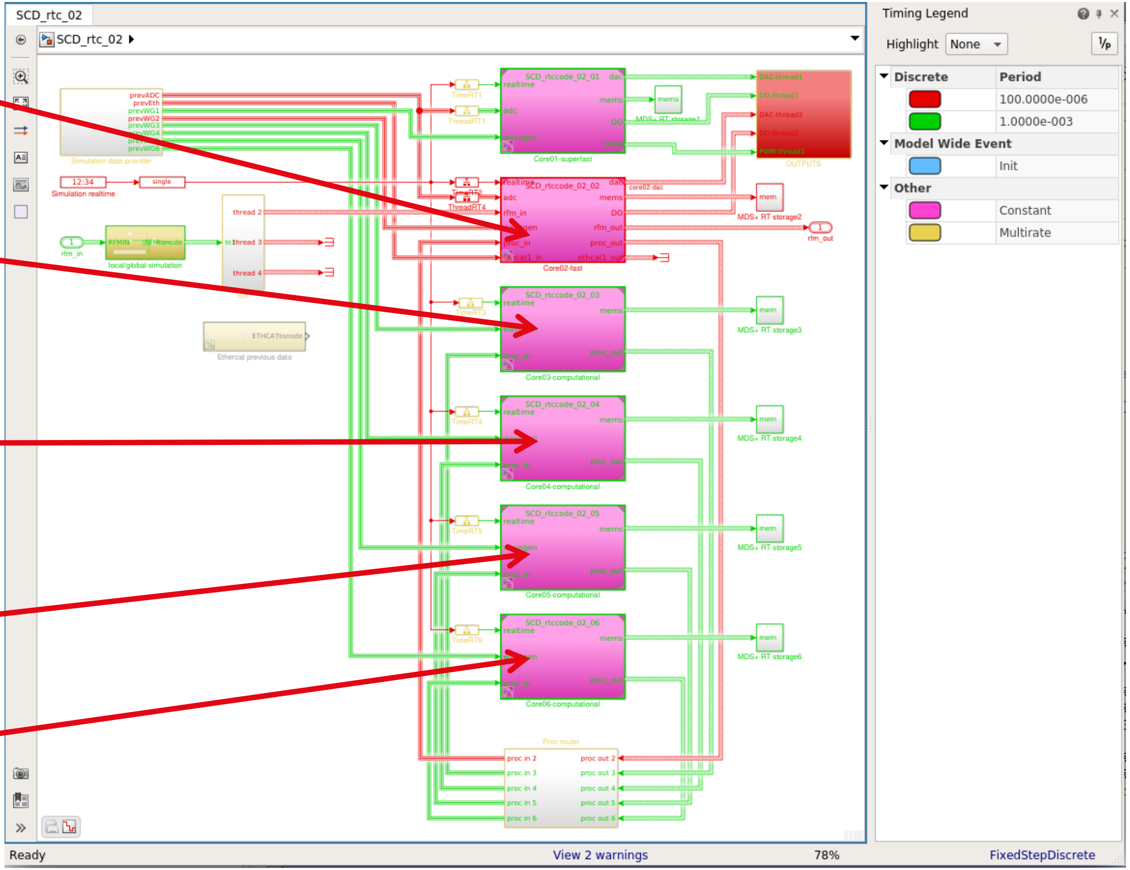
- Actuator manager and pulse scheduler (SAMONE)

CPU#5

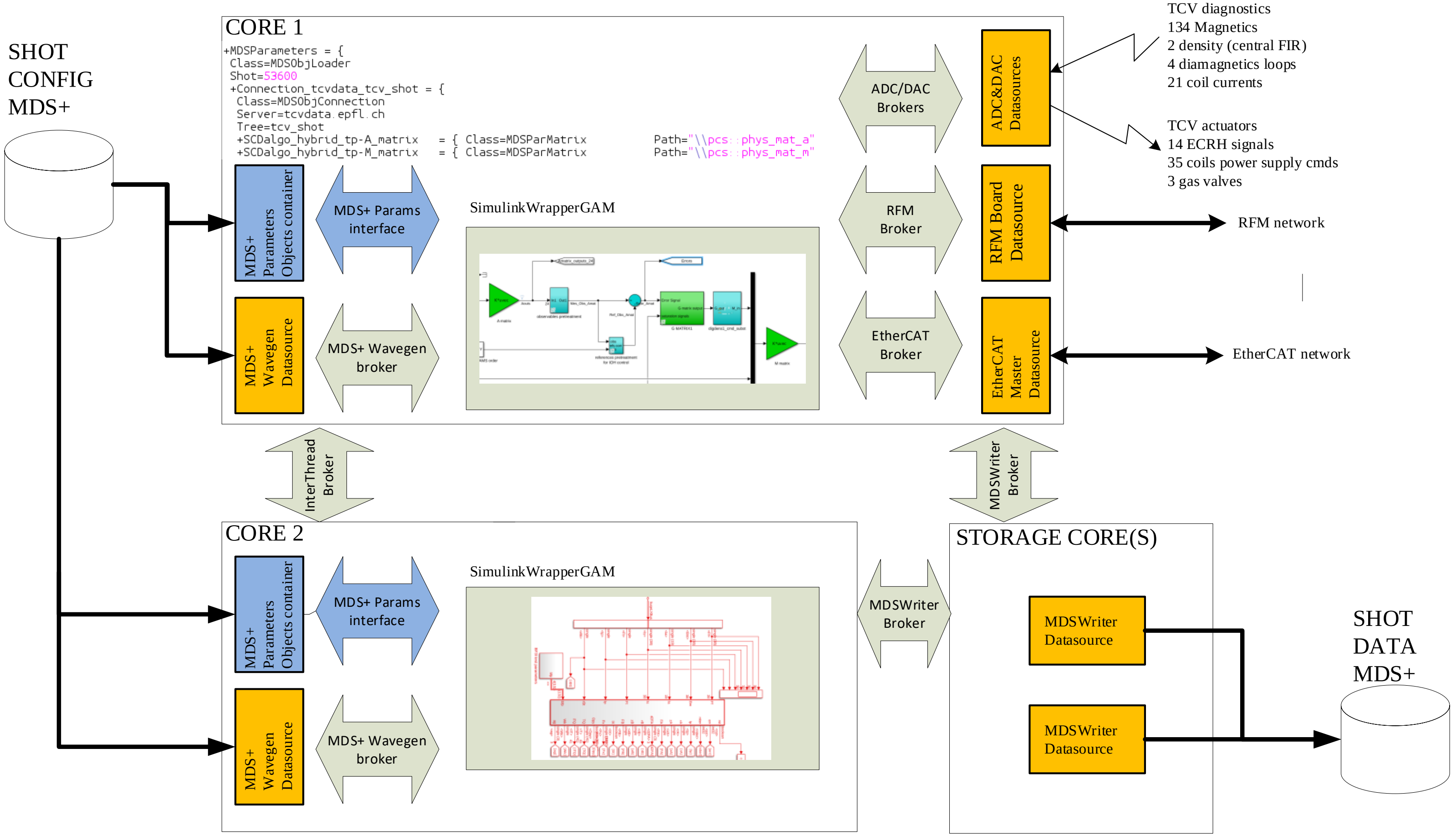
- RAPTOR

CPU#6

- RAPDENS

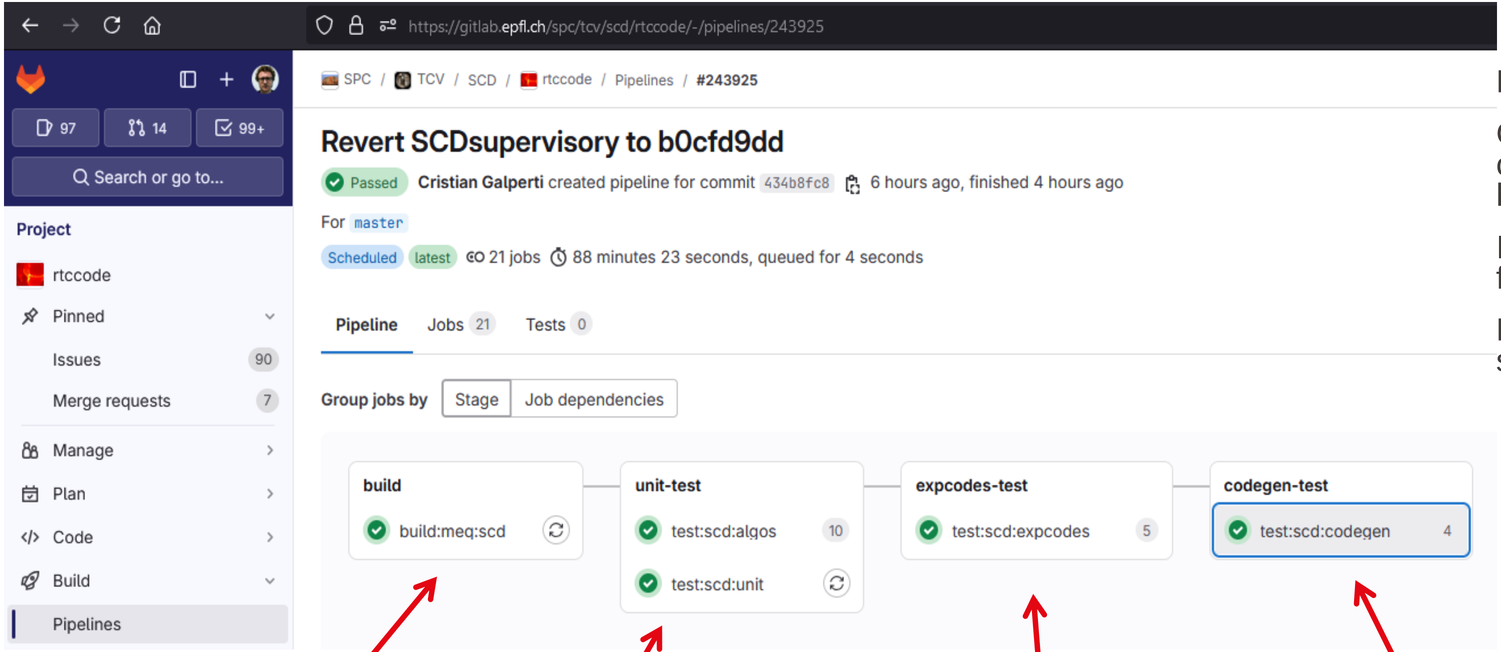


And its deployment with MARTe2



Galperti Cristian

CI/CD aspects, test pipelines



Directions:

Generated experiments code test on real control hardware

Initially, open loop against fiducial datasets

Final goal, closed loop simulations

...

Pre-requisite libs tests

- meq

Unit tests

- user algorithms harness tests
- Toolboxes tests

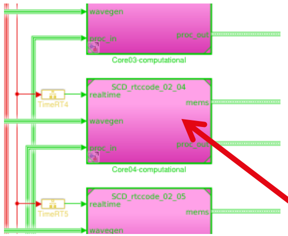
System wide matlab tests

- Whole control experiments code open loop simulation tests

Code generation tests

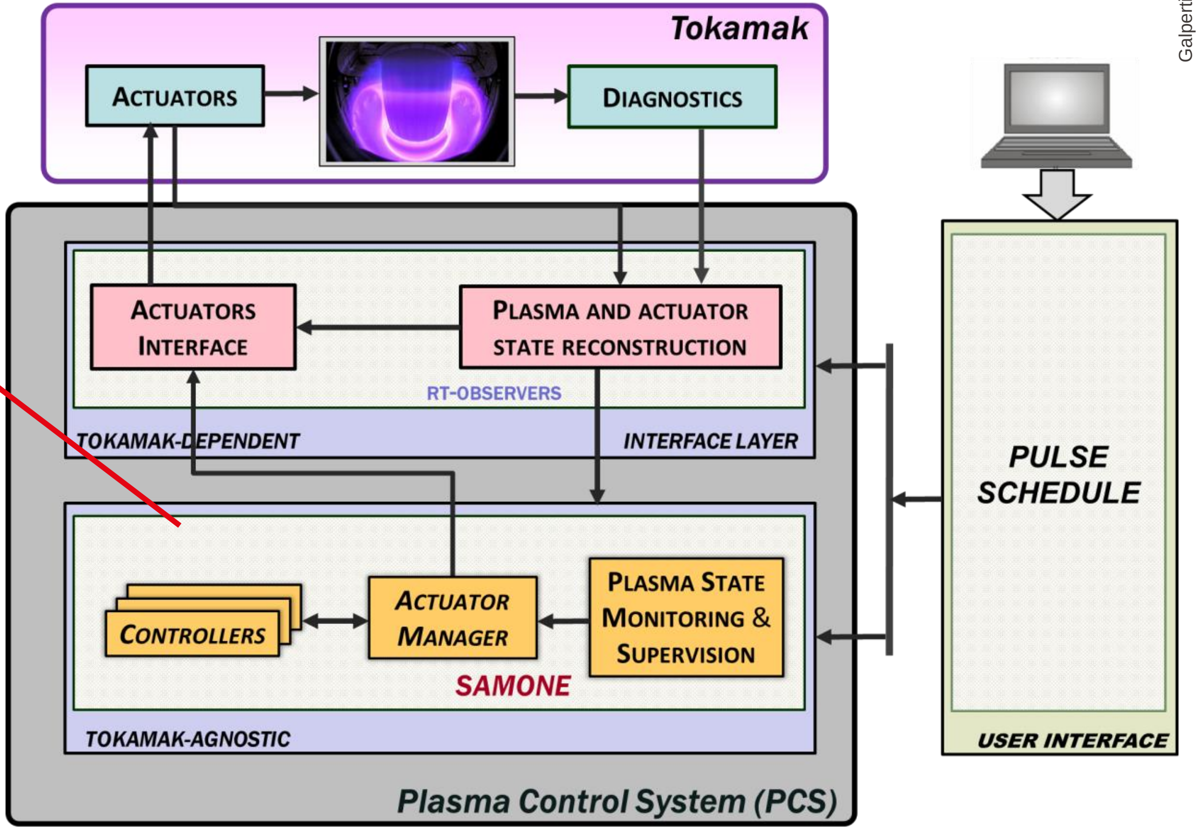
- Whole control experiments code generation tests

CPU #4 SAMONE (supervisory actuator manager)



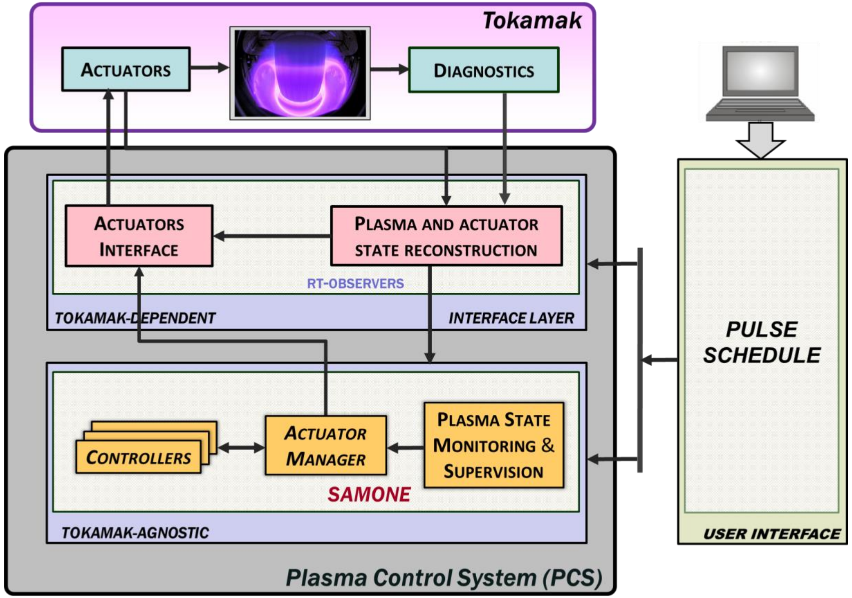
- Separation of tokamak **dependent** and **agnostic** layers
- **Generic** implementation
- Concepts of **integration** and **portability**

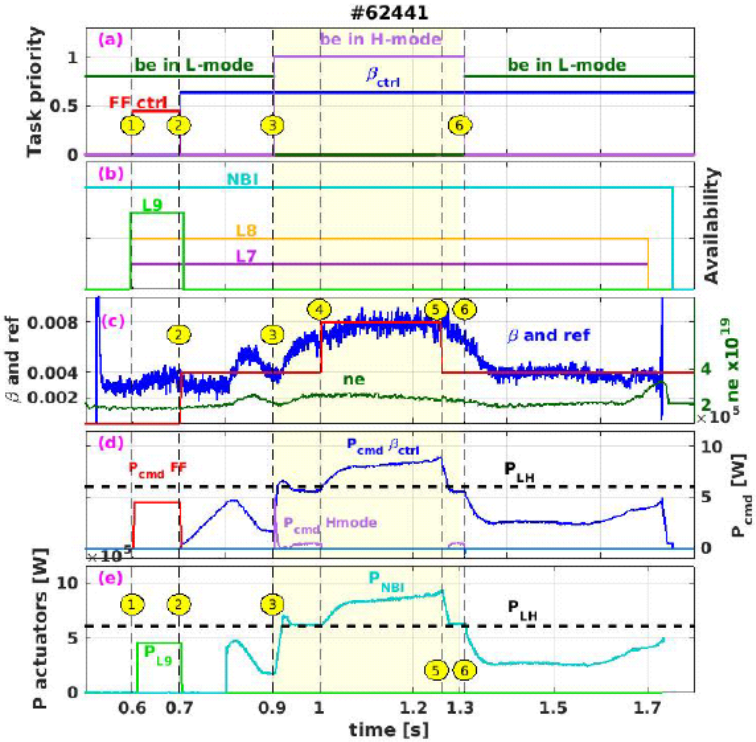
Lecture 10



REF: [T. Vu et al. IEEE TNS 2021]
REF: [F. Felici et al. IAEA 2021]
REF: [C. Galperti et al IAEA TM on Plasma Control Systems 2021]

SAMONE, two tasks: beta control + L/H status control



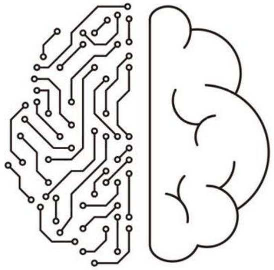


- Complete configurable plasma supervisory, actuator manager and off normal events handler to deal with multiple tasks and few actuators

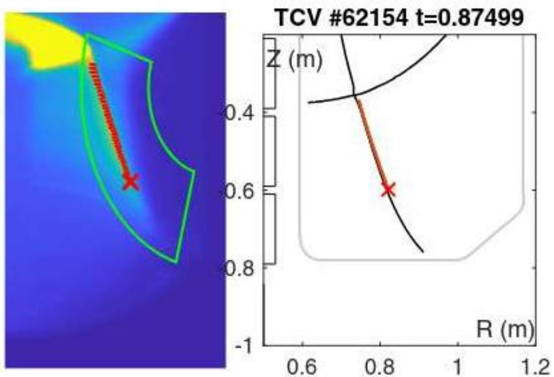
- Beta real-time control together with confinement status (L/H) control via NBH 1

REF: [T. Vu et al. IEEE TNS 2021]
REF: [F. Felici et al. IAEA 2021]
REF: [C. Galperti et al IAEA TM on Plasma Control Systems 2021]

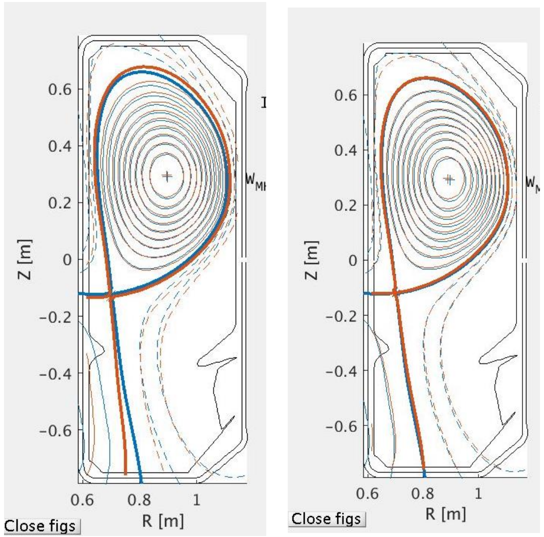
Cutting edge applications



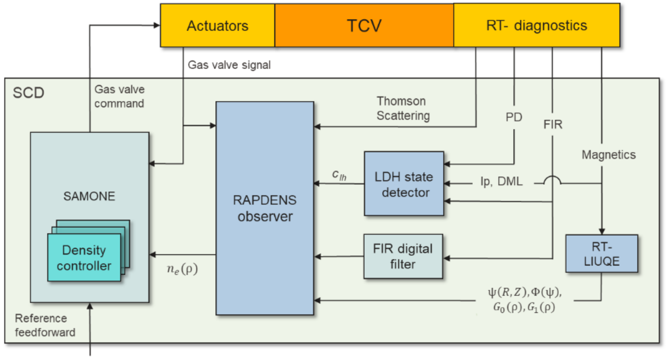
AI applications
Lectures 11a, 11b



Divertor heat flux control
Lecture 7



Active plasma shape control
Lecture 8b



Model based profiles and density control
Lectures 9a, 9b

Conclusions, outlooks, take home points

- Tokamaks control software and hardware is continuously evolving, and this is good since control research on them does the same.
- Keeping control systems updated is a key point for a tokamak plant, as they can accommodate newer ideas (e.g. machine learning) or more complex codes (e.g. transport simulators, microwave and neutral beams ray-tracers).
- There will be more and more need for globally, plant wide interconnected control systems as more and more complex control tasks will be put in their hands (disruption avoidance is a primary example).
- Advanced, interconnected control system will be of primary importance for existing and next future long pulse tokamak, ITER at first.

Thank you

cristian.galperti@epfl.ch