



Control loop with kepler and Simulink

P.Huynh, IM Design Team, CEA IRFM
Work done with JF.Artaud, F.Felici, O.Sauter



- One prototype of a IMAS-PCS coupling



- Use tools provided by EU ITM-TF/ISIP project:



- UAL (Unified Access Layer)
- Kepler framework



- Constraints:

- PCS Simulator implemented in Simulink and can be used as a C code
- Modularity, easy to insert a PCS
- Performance
- Interactivity

CHALMERS





- The control loop model is a step in the reference plasma current (I_p)
- We used an ITER test-case with hybrid scenario. The I_p control begins in the plateau phase.



- Plasma simulator is Metis



- It is a hybrid 1D/0D fast Plasma Simulator coded in matlab
- Has Input and Output from a structure "scenario" which contains among others:
 - the reference of Psi edge (poloidal flux at the edge), input via actuator simulator from the PCS
 - the value of Psi edge and I_p output from plasma simulator





- We demonstrate 3 possible implementations of the PCS Simulator as a Kepler Actor. The PCS Simulator actors have Ip in input and provide the voltage coils (Uext).

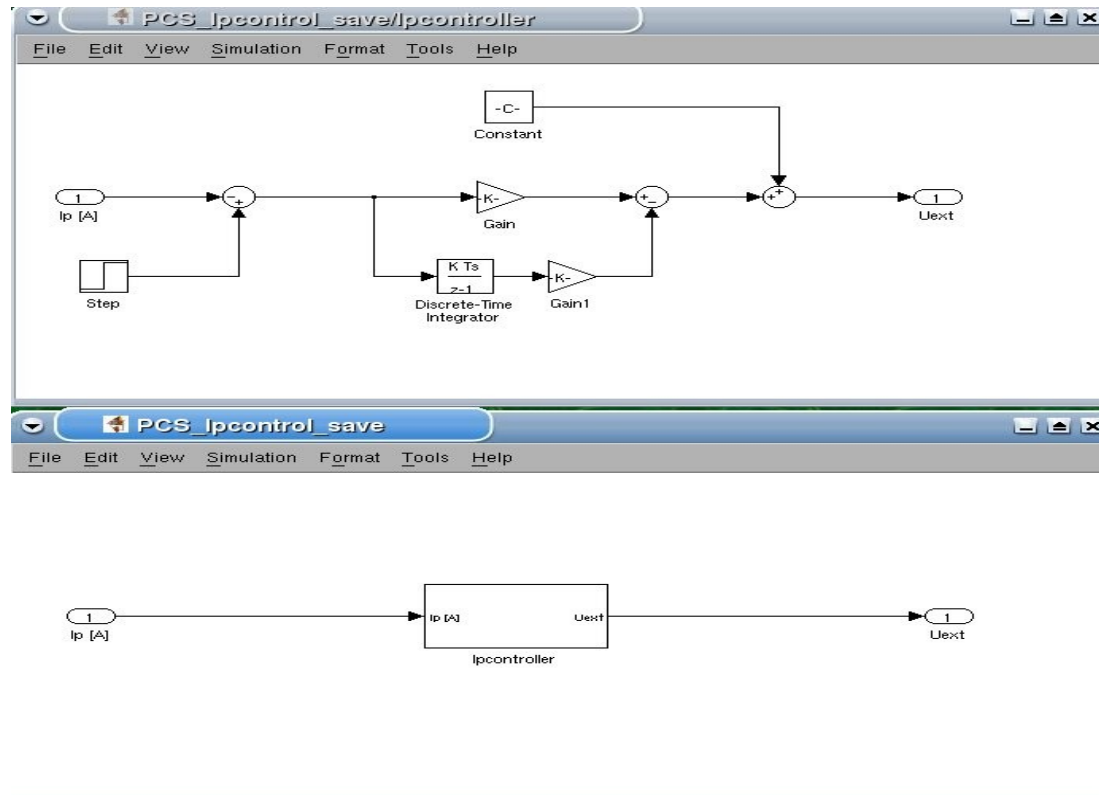


1/ matlab script

```
if time>402
    ipref = 14e6;
else
    ipref = 12e6;
end
kp = 3.e-6;
uext = kp*(ipref-ip)+0.60;
```



2/ C code auto-created from a Simulink controller



3/ run the simulink controller under matlab actor



CHALMERS





Components Data

Search

Search repository

Search Reset

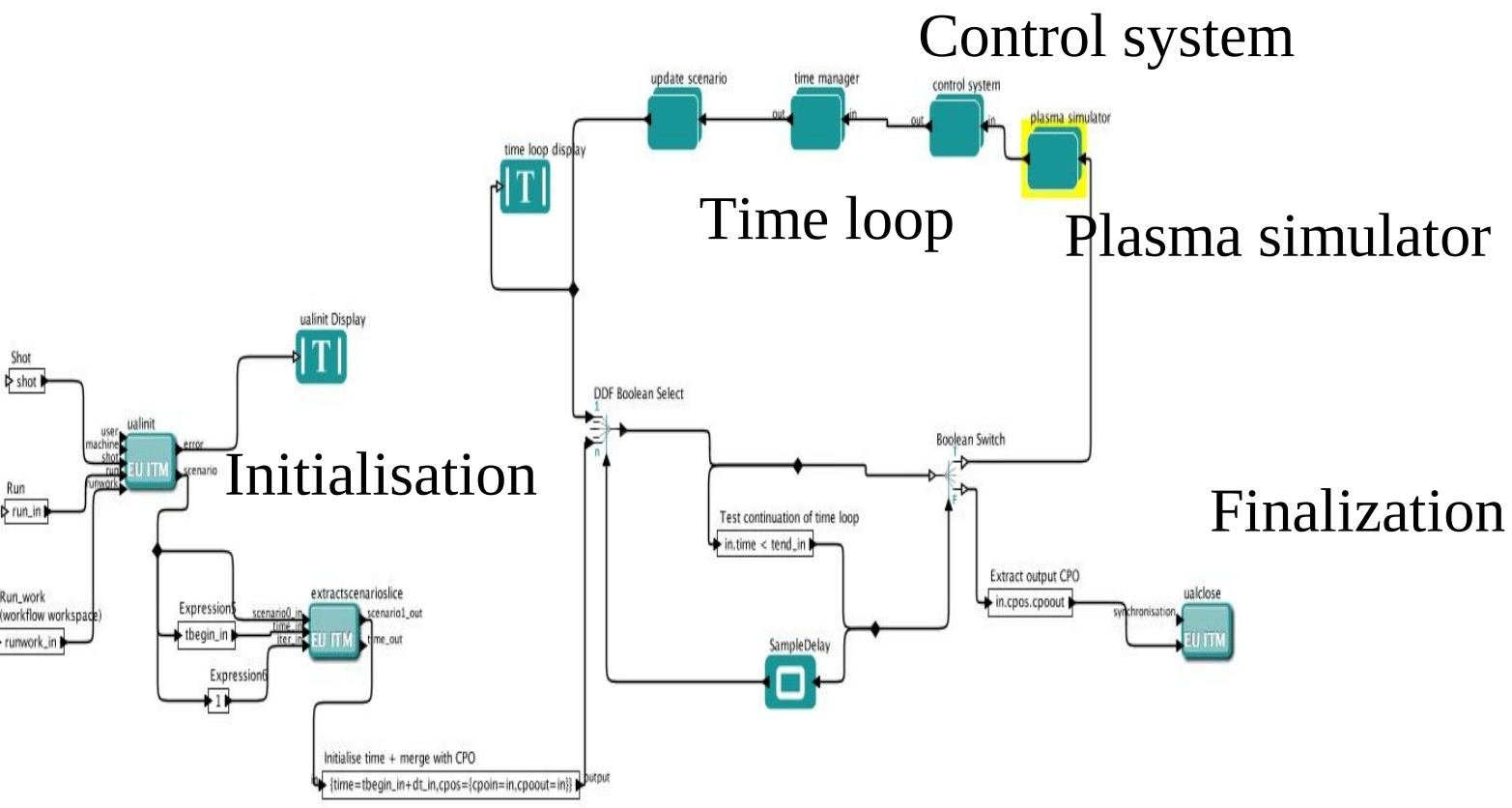
Components

Projects

Disciplines

Statistics

- shot: 5
 - run_in: 1
 - dt_in: 0.1
 - tbegin_in: 400
 - tend_in: 403
 - runwork_in: 7
 - runout_in: 3
- DDF Director
- Provenance Recorder



Control system

Time loop

Plasma simulator

Initialisation

Finalization

0 results found.



Components Data

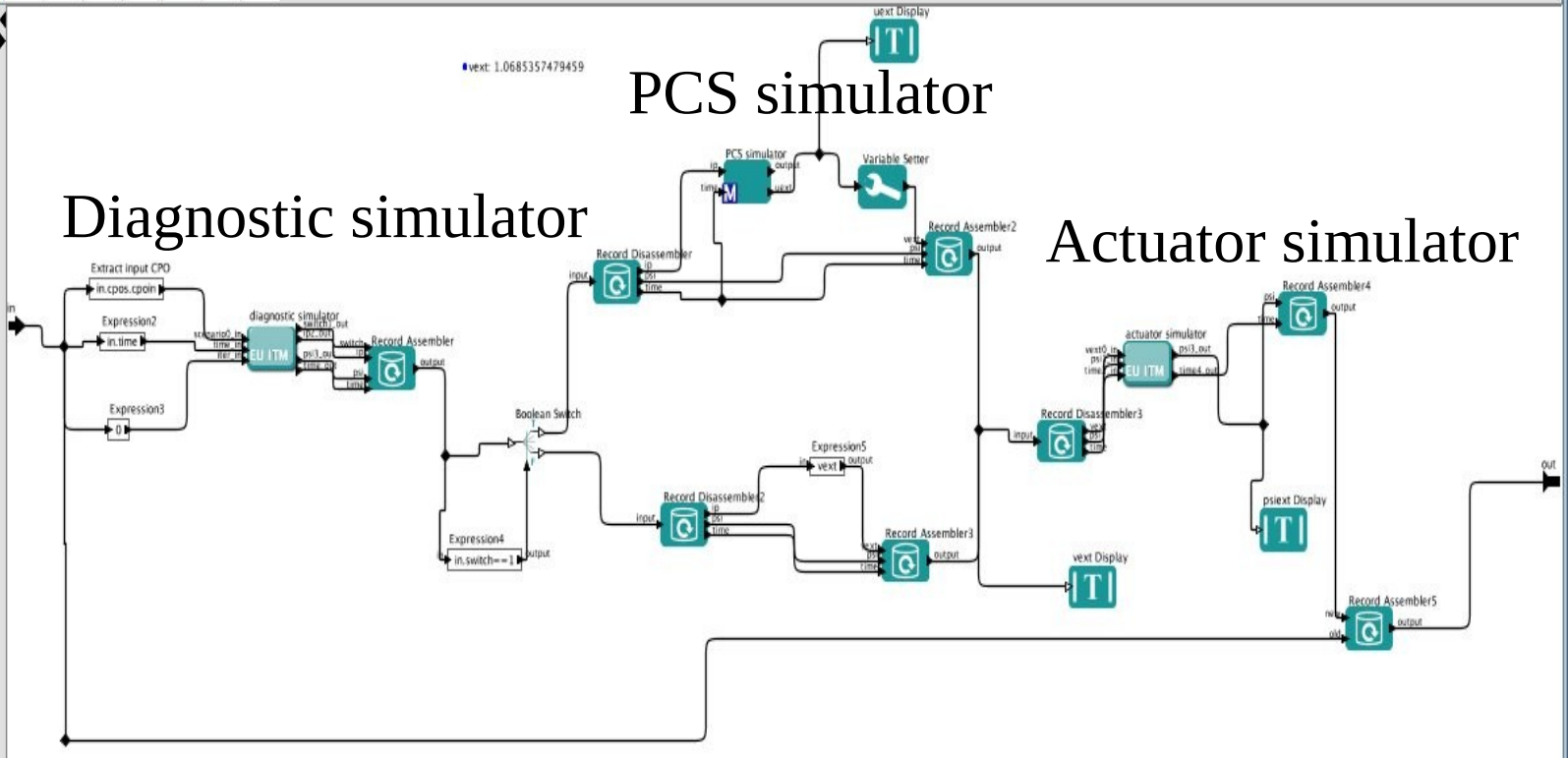
Search

Search repository

Search Reset

- Components
- Projects
- Disciplines
- Statistics

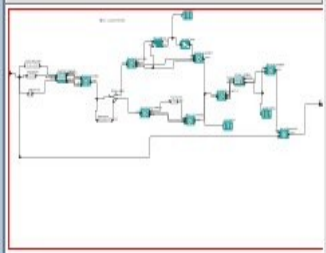
0 results found.



Diagnostic simulator

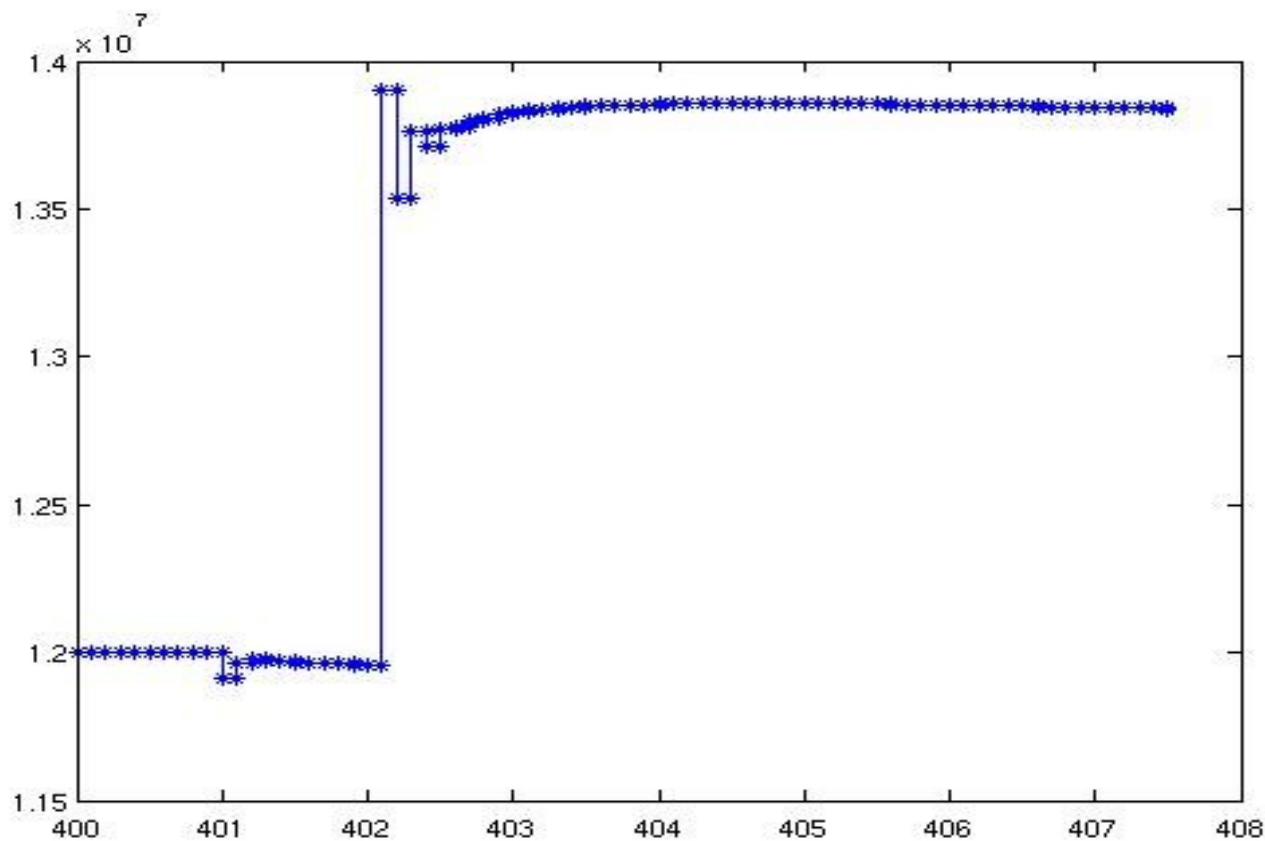
PCS simulator

Actuator simulator



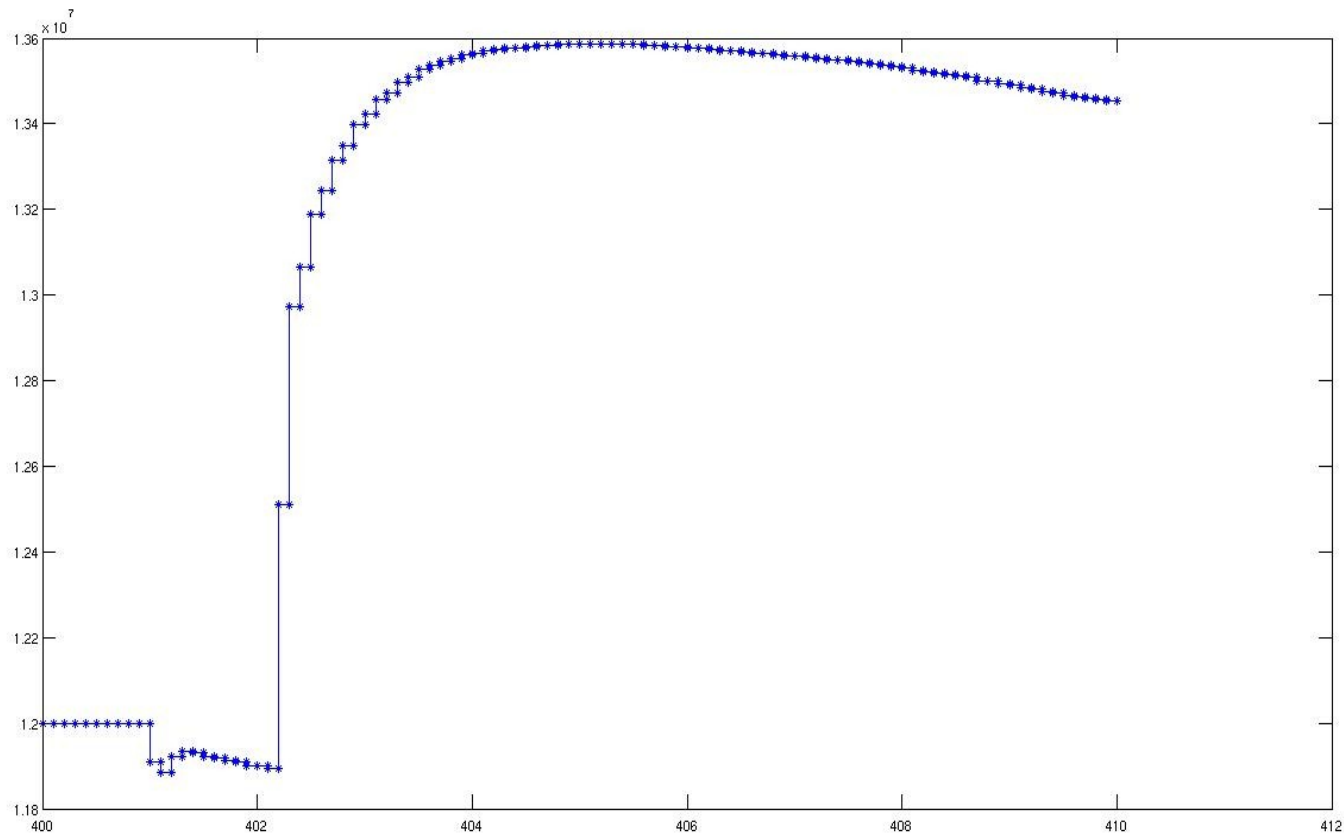


- With matlab script, $K_p=6.e-6$, the acquisition period the same as the time step



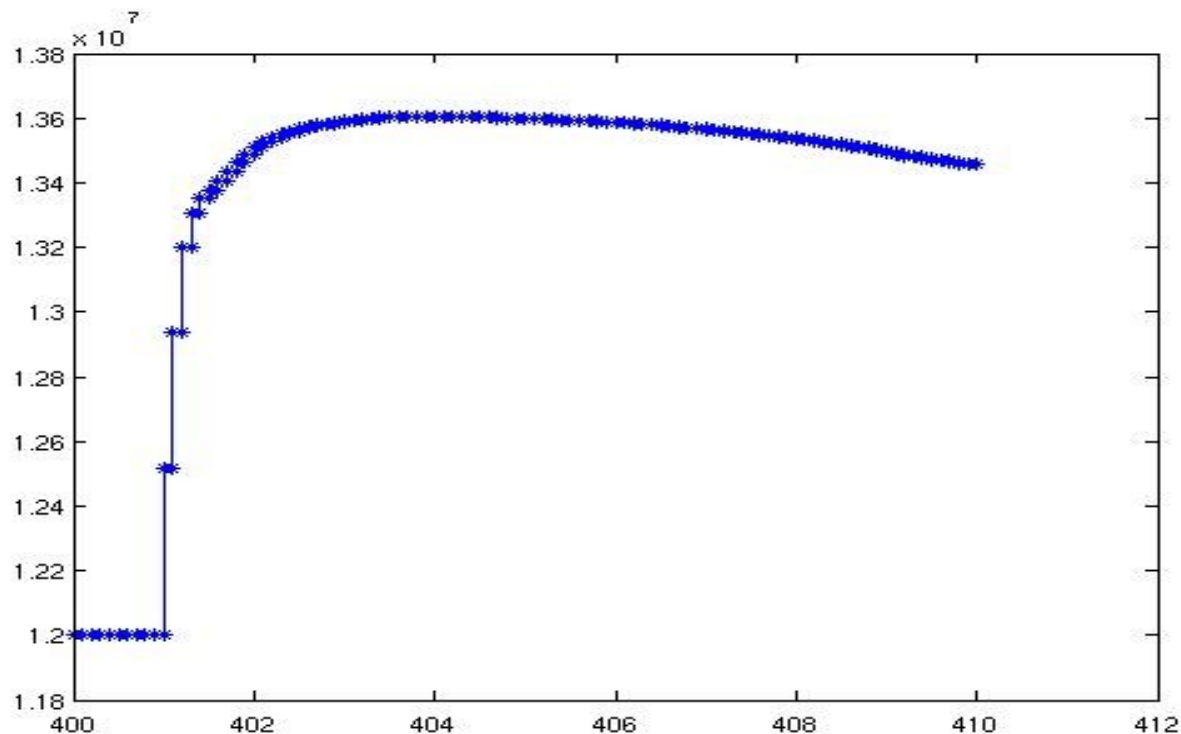


- With matlab script and acquisition period 2 times higher than the time step





- With C code from the simulink schema and a begin time of the step at 401s



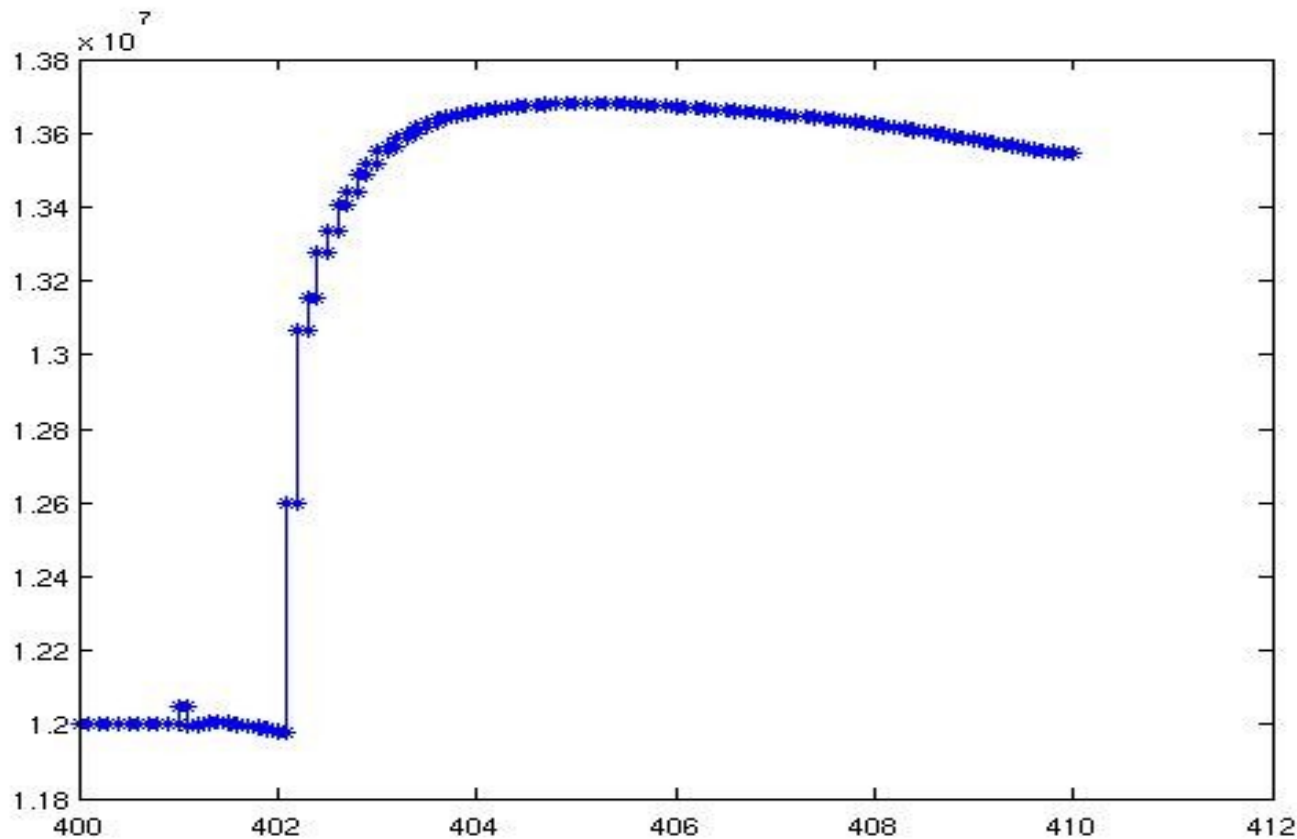


•

With the simulink schema and acquisition period 2 times higher than the simulator time step



CHALMERS





- Kepler can manage PCS as a simulink code or as a simulink-generated C code

- Modularity

- With a standardized list of Input and Output for the PCS, it can be easy to change from one PCS to another

- Performance

- Workflow can be executed concurrently for instance execute multiple PCS
- Can run efficiently with C code version (as for tokamak application)





•Interactivity

– Interaction with the workflow

- Kepler can control the workflow execution (start/resume, pause, stop)
- Kepler can deal with conditional breakpoints
- During the pause, you can change the Kepler parameters

– Interaction with PCS for developing purpose

- You may want several type of PCS execution: Run, Continue, Continue with a new simulink schema and the same internal state, Continue with a new simulink schema but with a new initialization
- You want to debug by using scope
- The implementation of such interactivity is possible mainly because the simulink schema is interpreted and the flexibility of simulink
- The implementation of such interactivity can be obtained quickly with a simple solution (file based) with few integration in Kepler. For an increased integration inside Kepler, it needs more work.

•Development of either "IMAS" or "PCS" has been demonstrated with the same tokamak simulator

