

## I. RAPID CONTROL PROTOTYPING AND LINUX RTAI

Rapid Controller Prototyping (RCP) requires two components: a Computer Aided Control System Design (CACSD) software and a dedicated hardware with a hard real-time operating system. The software should allow to perform all phases of control system design (specification, modelling and identification, control design, simulation, implementation and verification) in the same environment. A graphical application allows the user to create models for dynamic systems simply by connecting blocks from available libraries. Among others, some blocks implement linear systems given as transfer functions or state-space realizations both in continuous and discrete time. An integrated code generator allows to directly create code from the graphical scheme. The COMEDI project ([1]) provides the drivers to interface the system with the real plants.

### A. RTAI-Lab

The proposed solution is based on Linux RTAI, a hard real-time extension of the Linux Operating System developed at the Dipartimento di Ingegneria Aerospaziale del Politecnico di Milano (DIAPM) ([2]).

Linux RTAI provides a tools called RTAI-Lab ([3]), which contains a set of utilities for the integration of two CACSD software, the commercial Matlab/Simulink/RTW and the open source Scilab/Scicos suites, into the Linux RTAI environment.

### B. Matlab/Simulink/RTW

The Matlab/Simulink/RTW suite [4] is a commercial product widespread in universities and industries. Near to an accurate graphical interface it gives the possibility to create C code, using the RealTime-Workshop Toolbox. The generated code can be easily adapted and downloaded to different targets.

The core of the implementation for the Linux RTAI environment is represented by the 2 target files "rtai.tlc" and "rtai.tmf" and by the different C-MEX S-Functions that implement the data transfer to the RTAI-Lab environment and to the drivers of the DAQ boards (COMEDI or user specific). Fig. 1 shows the current Simulink RTAI Library.

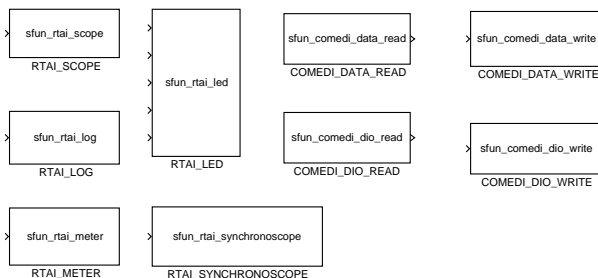


Fig. 1. Simulink RTAI Library

The Matlab/Simulink/RTW suite allows to generate code from a graphical scheme which may contain continuous blocks. In this case a numerical integration algorithm is

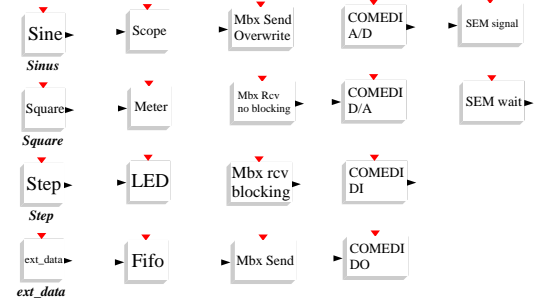


Fig. 2. Scicos RTAI library

added to the generated code. Multiple sampling rates can be handled by starting a real-time thread for each sampling time.

The main disadvantage of this suite is given by its cost (this solution requires licences for Matlab, Simulink, Control System Toolbox and Real-time Workshop). Furthermore the last release of the suite needs a high amount of system resources.

### C. Scilab/Scicos

Scilab ([5]) is a scientific software package for numerical computations providing a large set of functions for engineering and scientific applications. It has been developed since 1990 by researchers from INRIA (Institut National de Recherche on Informatique et on Automatique, [6]) and ENPC (École Nationale des ponts et chaussée, [7]) and it can be freely downloaded from the Internet. Scilab is currently used in educational and industrial environments around the world.

Scilab already contains all the functions and toolboxes needed for control applications. A tool called "Scicos" allows the implementation of block diagrams in a graphical mode. The simulation of the designed scheme can be performed directly from the Scicos window.

The RTAI environment has been extended with the new code generator *RTAICodegen\_sci*. This is a modified version of the *CodeGeneration\_sci* file provided by Scicos.

A new Scicos block library specific for the RTAI environment has been implemented. Fig. 2 shows the blocks contained in this library.

The RTAI library *libsciblk.a* contains the code of the Scicos blocks which need specific RTAI resources.

The code generator can only work with discrete time blocks. Multiple sampling rates must be implemented using distributed controllers. Asynchronous systems can be realized using semaphores or blocking mailboxes.

The main advantage of the Scilab/Scicos suite is given by the fact that all the software can be freely downloaded from the web.

### D. Code generation

A simple example shows the three distinct steps needed to create a stand-alone executable for the controller.

1. *Model definition:* The designer creates an appropriate Simulink or Scicos model using blocks from the standard built-in libraries (see Fig. 3 top). The plant is implemented as mathematical model including the non linearities of the system (Fig. 4 top). Using this scheme the designer can implement and simulate different control strategies and algorithms.

2. *Code generation:* The second step involves the C-code generation. The mathematical plant is now substituted by specific I/O blocks in order to implement the interfaces to the acquisition boards (see Fig. 4 bottom). Scopes must be substituted with the ones provided by the RTAI-Lab libraries (Fig. 3 top). In Matlab/Simulink the code generation is performed by the RealTime Workshop toolbox (RTW). In Scilab/Scicos the user must create a superblock first, and then he can start the code generation by calling the new provided function "RTAICodeGen.sci".

3. *Code execution:* Finally, the stand-alone executable can be directly started or transferred to any target machine running the same version of RTAI used for compilation.

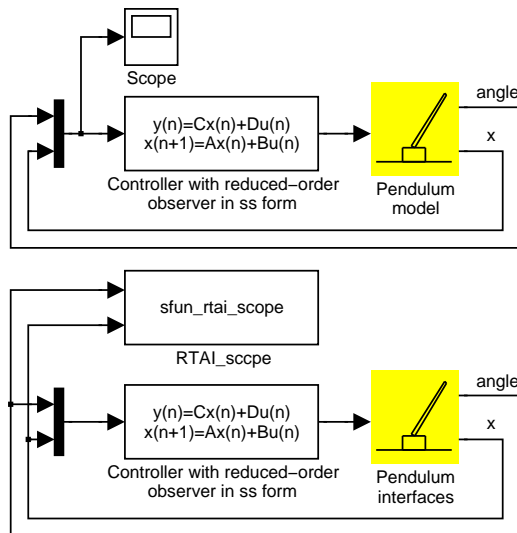


Fig. 3. Scheme for simulation (top) and code generation (bottom)

This procedure is similar in both Matlab/Simulink/RTW and Scilab/Scicos environments. Both systems contain a main file called *rtmain.c* which starts the realtime thread for the control task. The main file starts also a communication thread between the realtime task and an external GUI monitoring application (xrtailab or rtai-xml).

#### REFERENCES

- [1] COMEDI. Linux Control and Measurement Device Interface. [Online]. Available: <http://www.comedi.org>
- [2] RTAI. Real Time Application Interface. [Online]. Available: <http://www.rtai.org>
- [3] R. Bucher and L. Dozio, "CACSD with Linux RTAI and RTAI-Lab," in *Real Time Linux Workshop*, Valencia, 2003.
- [4] Mathworks. The MathWorks. [Online]. Available: <http://www.mathworks.com>
- [5] Scilab. A Free Scientific Software Package. [Online]. Available: <http://www.scilab.org>

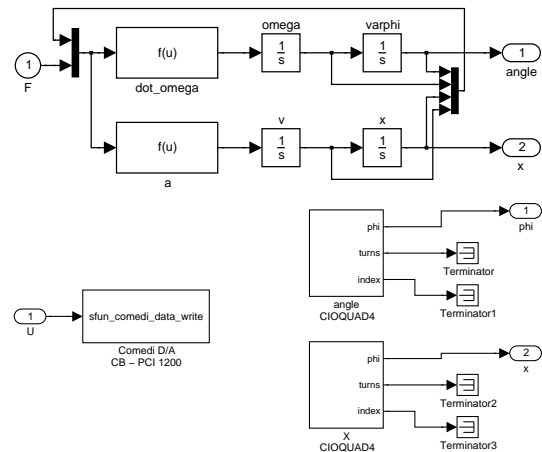


Fig. 4. Pendulum block for the simulation (top) and code generation (bottom)

- [6] Inria. Institut National de Recherche en Informatique et en Automatique. [Online]. Available: <http://www.inria.fr>
- [7] Enpc. École nationale des ponts et chaussées. [Online]. Available: <http://www.enpc.fr>