User Manual

DCU
CONTROL SYSTEM
PLATFORM
–
OVERVIEW

PROSYSTEMY, s.r.o.
Infoline@prosystemy.sk
www.prosystemy.sk

User Manual

DCU Control System Platform – Overview

07/04/24

Contents

1 Introduction

Dynamic control unit (DCU) Control System Platform is a complete set of software and hardware tools to build control system. DCU Control System Platform includes:

- Software for design and programming real-time control algorithms (Scilab/Xcos or Matlab/Simuling + DCU Design Tool)
- Software for building monitoring and visualization of control system (JAVA applications, Apache/PHP + MySQL + web-pages) sometimes called Supervisory Control and Data Acquisition (SCADA) system
- Real-time controllers (Dynamic Control Units) executing real-time control algorithms (DCU56IO, DCU28IO).

What distinguishes DCU Control System Platform from other control systems is accumulation of advanced and user-advantageous features from different types of control systems - scientific, industrial, and hobby:

- Scientific platforms bring optimal performance, robustness, and quality of control. Scientific control algorithms are used including advanced design techniques such as dynamic simulation or controller optimization. Scientific computational platforms are used for control design.
- Industrial features bring practicality, ease of implementation, user and designer comfort, flexibility. All typical industrial requirements on process control system are implemented – process data recording, graphical design and monitoring tool, user visualization, networking, user access management, security...
- Hobby features bring affordable price, openness, flexibility, independence. Only open-source and free software is used and multiple operating systems Win/Mac/Linux/Android are supported. Used software platforms are confirmed actively developed high-performance solutions.

Thanks to this selection of advanced features DCU Control System Platform is highly flexible and scalable as well as powerful. User can build a simple low-cost stand-alone embedded control, as well as special on-measure experimental or hobby set-up or a large and complex process control system (see Figure 1).

Structure of DCU Control System Platform is shown in Figure 2 and 3. The Figure 2 shows general structure of control system that can be build. The next figure displays internal software structure of the designer PC, or visualization server or operator panel. Control system is structured into three layers: real-time control, networking, and visualization/control design.

The visualization (SCADA) services may be executed separately on a stand-alone visualization server. Design software (Scilab or Matlab) is not needed in this case. However, visualization software (PHP, Apache,...) are useful to be installed on designer PC so he can prepare visualization on the same computer where control design is made. The middle-ware: Java Real Time Environment, MySQL database, and User Command Center (UCC) JAVA application are necessary for both tasks control design and visualization.

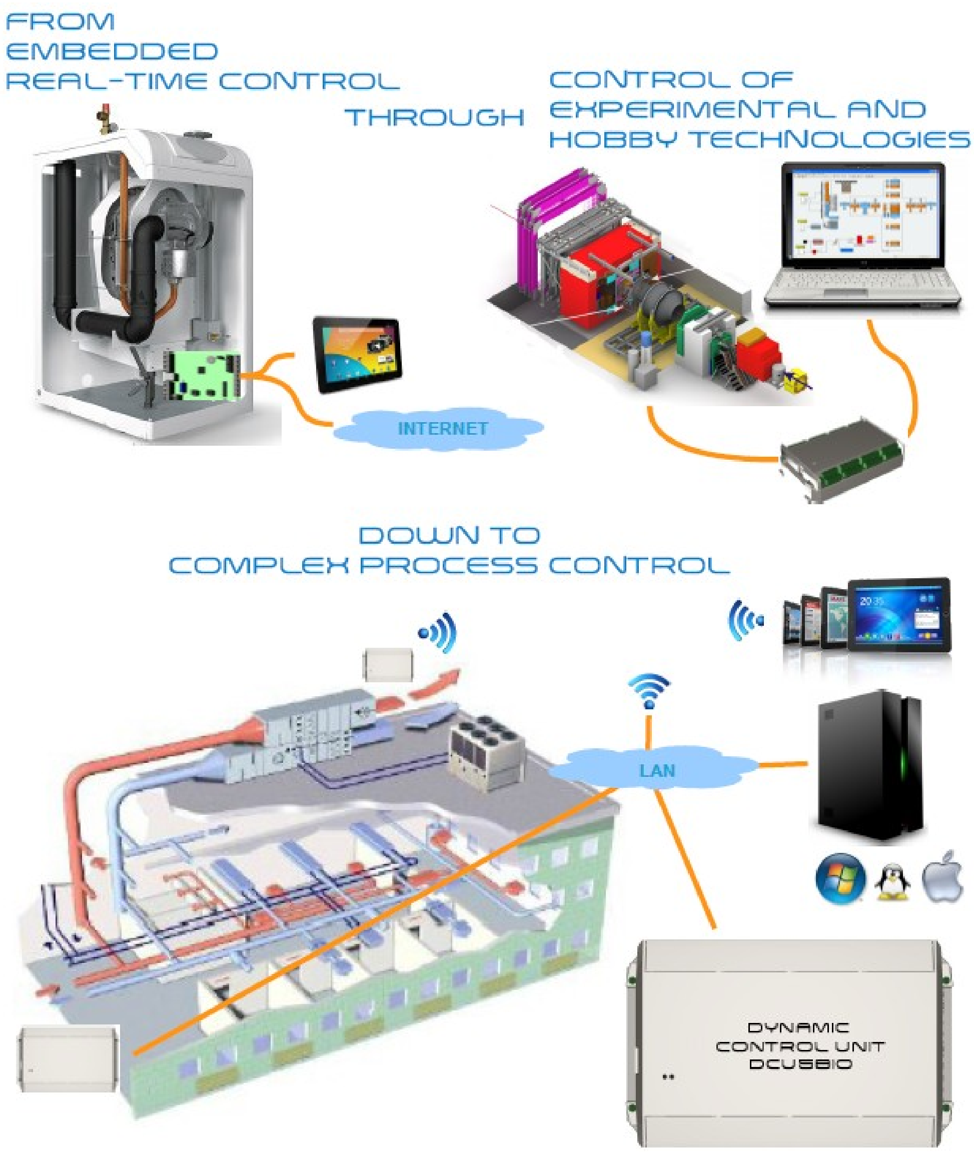

Figure 1: Application fields of DCU Control System Platform

2 System requirements

Supported operating systems for control design are Windows 32-bit or 64-bit, Linux 32-bit or 64-bit, Mac OS. It is recommended to have at least 2GB (optimally 16GB) of RAM memory and dedicate to control design platform (Scilab or Matlab) at least 1GB (optimally 8GB). Large control applications (more than 300 control functions and variables) may require up to 12GB. More memory dedicated to control design platform also accelerates opening/closing/modification/simulation of control application diagrams. Java Runtime Environment (JRE) version 8 and higher must be installed (done automatically by installer).

Supported operating systems for visualization server and operator panels are Windows 32-bit or 64-bit, Linux 32-bit or 64-bit, Mac OS, Android 4.1 (Jelly Bean API level 16) and higher.

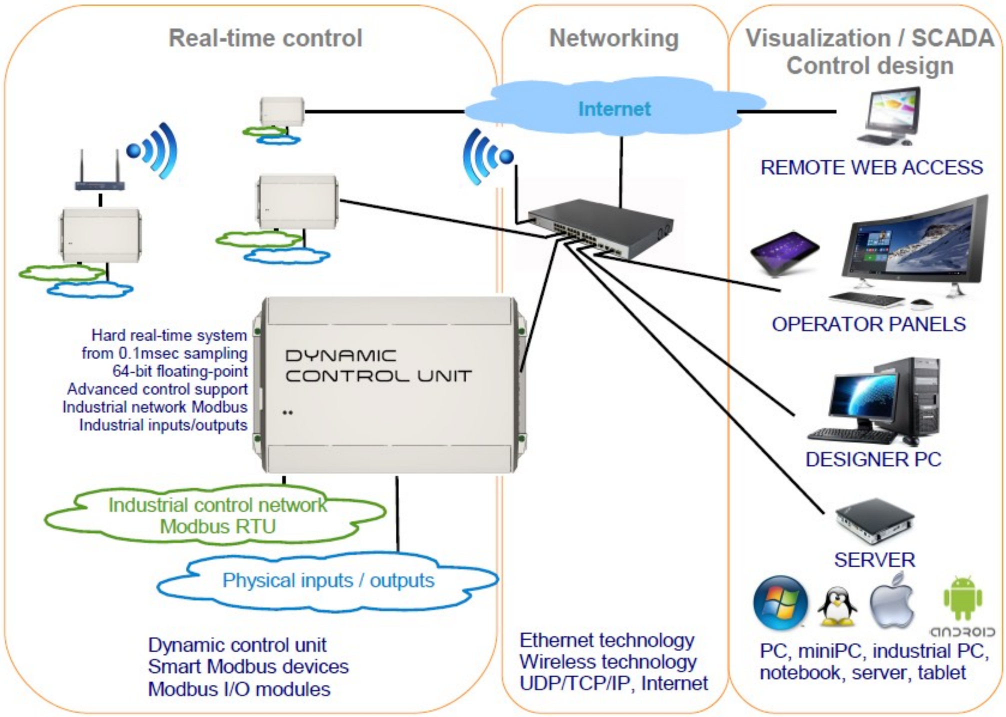

Figure 2: General structure of DCU Control System.

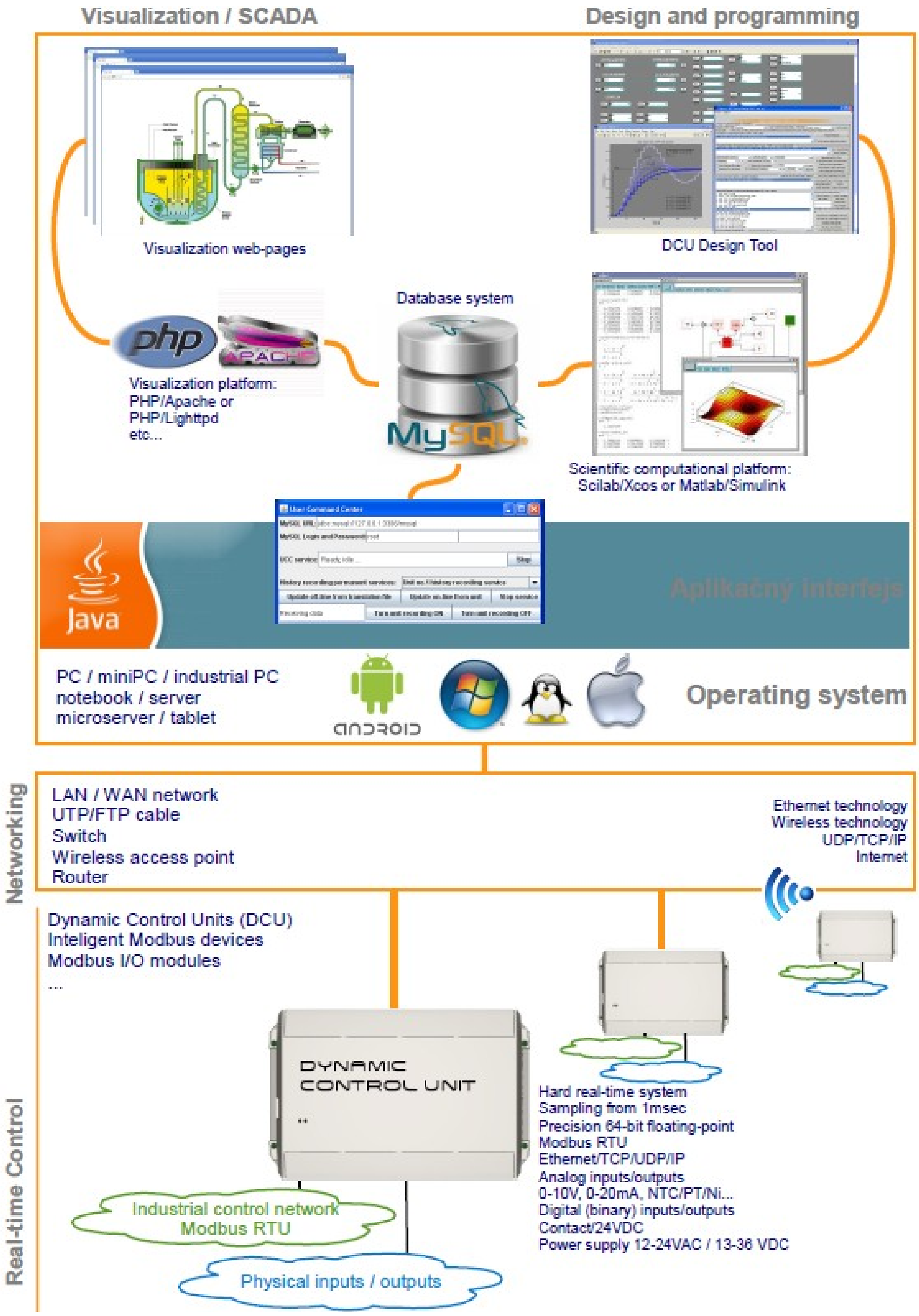

Figure 3: Structure of DCU Control System Platform.

3 Networking and communication capabilities

DCU Control System Platform implements two communication standards – Ethernet/TCP/UDP/IP and RS485 serial Modbus RTU, see Figure 4.

DCU physical connectors. DCU has 2 physical connectors:

- Ethernet RJ45 connector for connection to 10/100 Ethernet LAN network. This connector serves both real-time UDP/IP and TCP/IP protocols.
- RS485 serial link terminal provide physical connection to Modbus RTU bus.

DCU process control communication. Control system network interconnects control devices in order to transport process-control data mainly process variable values. DCU units may be interconnected using the following network protocols

- Real-time UDP/IP protocol
- MODBUS RTU protocol on RS485 serial line

DCU user communication. Designer PC and monitoring and visualization server are connected to DCU via TCP/IP protocol.

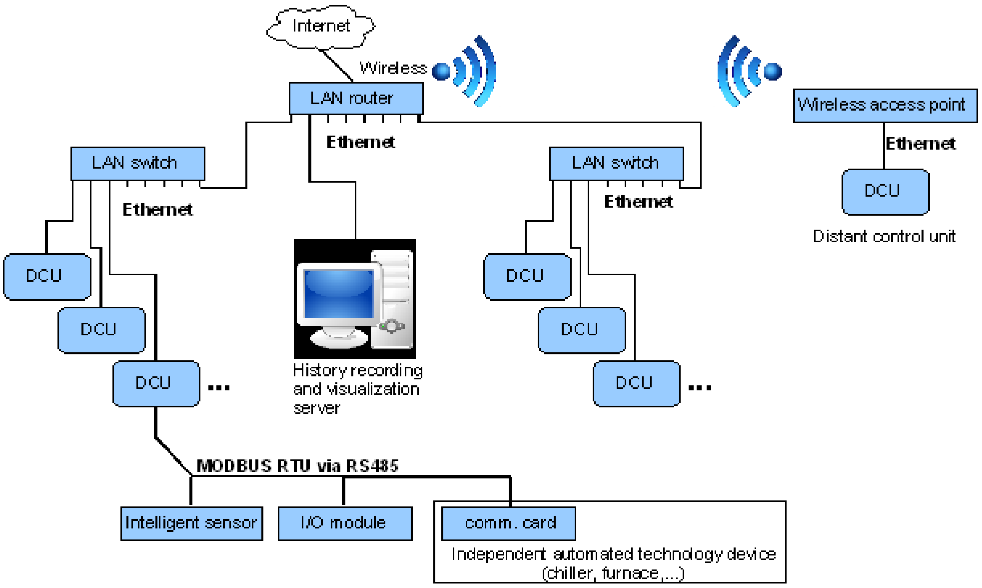

Figure 4: DCU networking.

3.1 Real-time UDP/IP protocol

For transferring process control variables between DCU controllers, UDP/IP protocol is envisaged. In selected sampling time, data are transferred from source DCU/server to destination DCU/server. Since UDP/IP protocol is used, data can pass through local network, wireless network, as well as global network (Internet). To ensure high transfer speed and to avoid delays and data loss, transfer critical data only via local network specifically built for control purposes.

3.2 MODBUS RTU bus

Standard MODBUS RTU industrial bus is implemented for connection with other producers control devices that use this standard industrial bus. DCU controller may be set as a master or as a slave. Serial communication settings (baud rate, parity bit, number of stop bits,...) are set inside control application. Typical examples of devices connected using MODBUS are

- MODBUS input/output module. DCU control system may extend physical I/Os without adding supplementary DCU unit.
- MODBUS communication card of an independent autonomous technology device (chiller, furnace, HVAC unit, etc.) that features its own control system. DCU control system may monitor state and performance of the device.
- Intelligent sensor/control panel/actuator with MODBUS connectivity

Note: *It is recommended to use separate power supply for DCU and other power supply for MODBUS devices and connect grounds of RS485 serial connectors. Use of termination resistor for RS485 serial link is recommended as well. See section on DCU electrical connections.*

3.3 User TCP/IP protocol

DCU controller communicates with designer PC and visualization server using TCP/IP protocol. Communication may therefore pass over local network, wireless network, as well as global network. Designer PC or server must have running User Command Center (UCC) JAVA service which performs actual direct communication with DCUs. To access data received by UCC service, MySQL database service is used. MySQL database system have to be installed and running on communicating PC as well.

Developer note: *For direct access to DCU, user application register request to UCC database called "dcuUserCommdb" into request table, which is regularly checked by UCC. When UCC receives request it decodes its meaning and parameters, performs the specified task, and returns result to the interface database into result table. Meanwhile, user application waits and regularly checks result table for its request result.*

For access to historical process data a user application may again use UCC service or it may access directly to data stored in the database. Each unit which has running history recording service has its own database of process data called "dcuHistRecdb_x" where x is ID number of DCU.

4 Real-time process control

Real-time control is primary executed by Dynamic Control Units. Real-time features include:

- Embedded hard real-time operating system.
- Double 64-bit computational precision.
- Functional diagram interpret (parser).

Hard real-time indicates that for any control algorithm programmed in the unit an exact timing (sampling time) may be set ranging from 1ms up to hours. Exact timing means that entire control algorithm cycle: reading inputs - evaluating - writing outputs is executed within the specified sampling time interval.

Double 64-bit precision means that any analog value used in any control algorithm inside DCU is considered to be either real 64-bit floating-point value (range $\pm 10^{308}$ with min. value 10^{-323}) or integer 64-bit value (range ±9,223,372,036,854,775,807). This precision is used and satisfactory for the most sophisticated control algorithms developed in scientific software tools Scilab or Matlab.

Functional diagram interpreter *is a part of hard real-time system. The interpreter executes control application functional diagram loaded into DCU. Notice that control application loaded into DCU has not a form of an assembler binary code to be executed. A functional diagram of control application is directly loaded into unit.*

5 Performance limits

Control application upload takes approximately between 30 and 90 seconds depending on control application size.

Maximums size of a control application is about 900 control functions.

Real-time system (firmware) upload takes at most 120 seconds.

Serving all 56 inputs and outputs requires about 1.2msec. Limit is imposed mainly by analog inputs, which has slowest communication rate.

A standard control application running in one thread, serving all inputs/outputs and with about 300 control functions and 300 control variables requires about 10msec of computational time. Minimal true sampling time for a larger application running in one thread and serving all 56 inputs and outputs is about 13msec. True sampling time for smaller applications serving only some inputs / outputs starts from 1msec. If unsatisfactory, control application execution speed may be optimized by distributing control loops into up to 3 computational threads. Loops requiring fast sampling may be placed into fast interrupt thread and remaining loops into medium or slow interrupt thread. For extremely fast control loops at 10msec sampling and faster, it is recommended to dedicate one small DCU that executes only that control loops.

Data recording from one DCU controller on low-cost mini-PC (Intel NUC Kit DCCP847DYE) was proved to work smoothly without any data loss at the rate of 160000 records/min. All 56 inputs/outputs of one DCU may be recorded into MySQL database with sampling of 25msec without loss of data. Note that database is supposed to be small. Performance of database system decreases with the size of database tables.

Scilab/Xcos simulation platform (version 5.4.1) proved to be slow for simulation of large control applications on a standard personal computer. In case of larger diagrams, the recommendation is to execute simulation of each control loop in a separate diagram and when satisfactory copy the loop into complete control application diagram.

Use of low-cost mini-PC (Intel/Asus NUC Kit) as a data recording and visualization server for handling 16 DCU56IO controllers with recording sampling rate 1sec proved to work without any problem. Linux Ubuntu 22 operating system was installed together with LAMP stack. Response of visualization system to operator action is about 3-4seconds. Response time may be easily improved by using more powerful hardware.

Control application is temporally interrupted in case of:

- Control variable override - single 90msec interruption (write to flash).
- Control application backup - multiple (tens) 3msec interruptions (write to EEPROM).
- Real-time system parameters modifications - single 50msec interruption (write to flash)
- Control function parameters modification - single 90msec interruption (write to flash).
- Control application load - multiple (tens) 90msec to 500msec interruptions (write to flash).
- Real-time system load - entire control application is stopped and erased, communication settings are reset to default (IP address reset to 10.0.0.10).

6 Visualization and monitoring

User monitoring and visualization (SCADA) system is expected to provide a comfortable interface for monitoring and operating technology or process. The following essential tasks are required from a monitoring and visualization system:

- Display actual values and states (alarm, override) for selected groups of control variables. The values and states may be displayed in a list or inside an image - a graphical representation of the controlled process. The values and states may be represented by numbers or text strings or images or animations.
- Display historical values and states of control variables. The same display configurations may be required by user as in the case of actual values to see process state in some previous moment. Signal graphs are needed as well to see evolution of control variable values.
- Display control unit events, such as control variable alarms, control unit failures, errors, initialization, etc.
- Set constant override (forcing or set to manual value) any control variable.
- Change a parameter of any control function.

Visualization and monitoring system consists, in case of DCU platform, from:

- Visualization server
- Operator panels (optional)
- Devices with web browser (PC, tablet, smartphone)

Visualization server is a computer with a package of software that receives and records process data from DCU controllers, sends process data to operator panels, send web-pages with process data to any web-browser. Visualization server of DCU Control System Platform consists of the following software

- MySQL database server for recording process data
- Apache http/web server with PHP (any other web server with PHP may be used) for presenting data in the form of web pages
- MySQL management tool phpMyAdmin. Any other MySQL management

tool may be used instead.

- User Command Center JAVA application connecting real-time DCU units with database. UCC requires Java 8 JRE installed.
- Web-pages of process visualization.
- Configuration databases that defines process visualization parameters.

Operator panel is a computer with installed visualization server software, but running in operator panel mode. In operator panel mode, computer receives most of process data from visualization server and not from real-time units. However, specific process variables are exchanged directly with real-time control unit to provide immediate interaction with process. Operator panel may display process data using its local web server or it can connect to server using web-browser (development discontinued).

Devices with web browser is any computer, tablet, mini-PC, smartphone with installed web-browser. Using the web-browser user/operator connects to visualization server to see process web visualization.

7 Control design and programming

DCU Control Design Tool is an application with graphical user interface that serves for programming, testing, and on-line monitoring of DCU (dynamic control unit) real-time controllers. The tool is programmed inside Matlab (Scilab) scientific computational platform using its specific language. The tool is one of many so called modules or toolboxes of this computational platform. Entire procedure of control design is not performed uniquely inside of the DCU design toolbox. Three main tools and its graphical interfaces are used for control design, see Figure 5.

The first design tool is DCU design toolbox with its graphical user interface, see Figure 5. It is used for project management, on-line monitoring, and control system management. User defines here control design projects, versions of the project, control application diagram translations, communications settings of translated real-time control units to connect on-line with the controller. Lists of control functions and process control variables of translated control units may be displayed. Units may be connected on-line, actual values of process control variables may be displayed. Process experiment tools may be launched over selected control functions or variables.

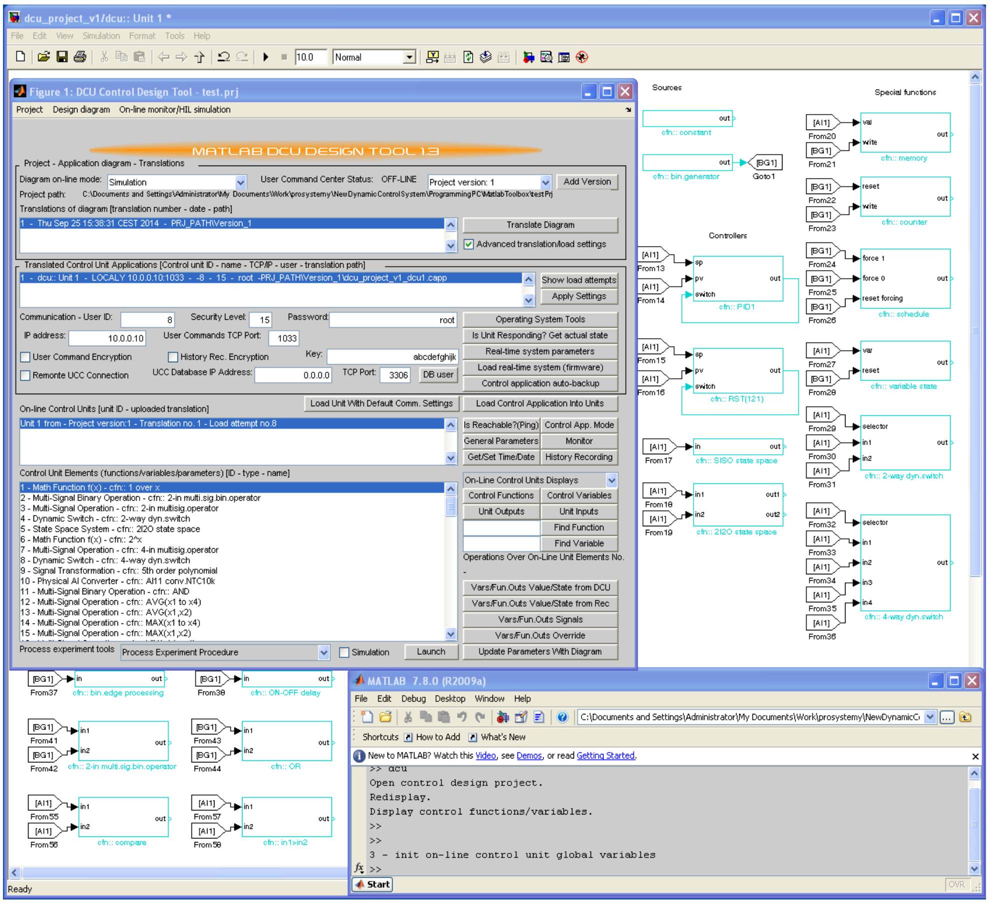

Figure 5: DCU control design toolbox main three tools.

Second tool used for control design is the Xcos / Simulink toolbox, see Figure 6. It is used for programming real-time control applications. Simulink/Xcos is a dynamic simulator with comfortable interfaces for drawing and simulating diagrams of dynamic models and in our case control application diagrams. User programs real-time control application by drawing functional block diagram of desired control application. Library of available real-time control functions is available containing all standard industrial as well as advanced control functional blocks. Since the diagram is drawn in the dynamic simulator, control application may be validated via dynamic simulations. User may define different operational scenarios, include dynamic process models, place displays and graphs all around of the control application and launch simulation. Dynamic simulation belongs to the most advanced validation techniques available.

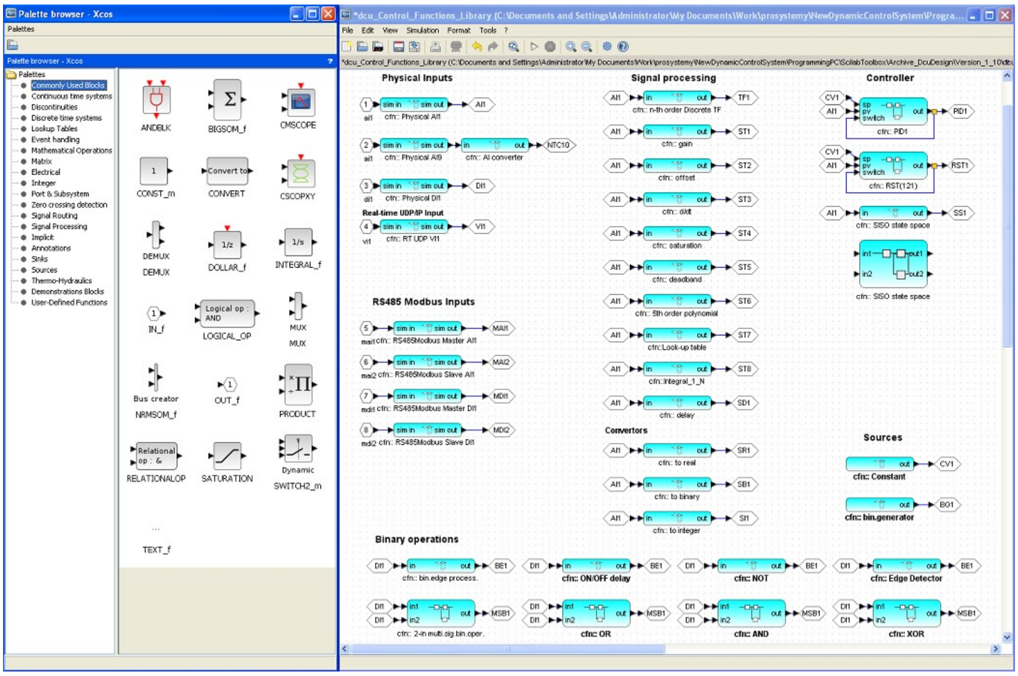

Figure 6: Graphical interface of dynamic simulator toolbox (Xcos).

Third design tool is the main command line window of the computational platform Matlab or Scilab (Figure 5 bottom right corner) and all available toolboxes that may be need for process experiments. For example, to prepare excitation signals, to analyze and display obtained process responses, etc. This third part of control design is optional and it is intended for experienced control systems engineers who require optimized and robust control system performance.