REAL-TIME CONTROL DESIGN TOOLBOX FOR DCU CONTROL SYSTEM

H. Prochazka

PROSYSTEMY, s.r.o., Slovakia

Abstract

A real-time control design and programming toolbox - DCU (Dynamic Control Unit) control design tool - is presented in this paper. The toolbox is a part of a new advanced fully-featured open-source based process control system developed at PROSYSTEMY. The tool serves for designing and programming of real-time controllers (Dynamic Control Units). Control application is in a form of Simulink diagram. Control application can therefore be interconnected with dynamic process models and simulated. Verified control application may be uploaded into real-time DCU controllers. The diagrams are uploaded, no conversion to C or assembler is performed. Uploaded control application is in consequence small in size and error-free. The toolbox provide tools for on-line process monitoring as well. The toolbox satisfies not only basic needs of a process control designer. Strong support of advanced control algorithms and techniques is implemented. For example a process experiment procedure is integrated that sends a signal into process in real time and collect process responses. Such procedure is necessary for many advanced design techniques (model identification, controller reduction, optimization) and it is nonexistent in standard control systems. A user may use this procedure directly or implement its own design technique into the toolbox and use the procedure inside of his code.

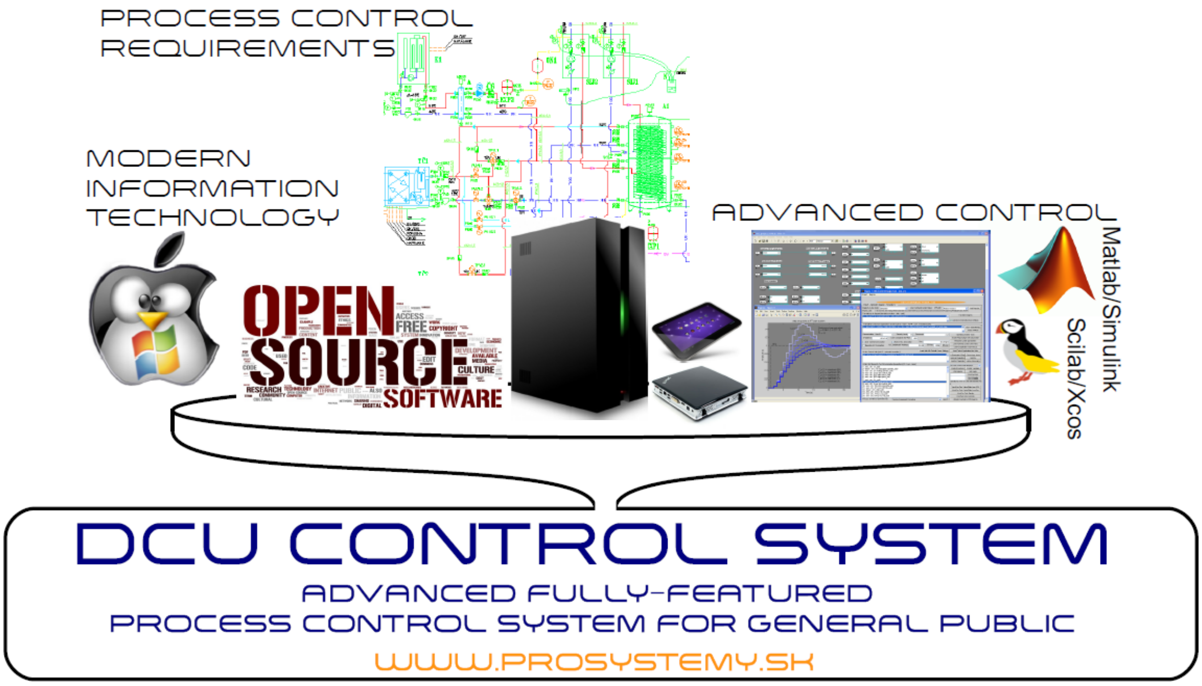

Figure 1: DCU control system

1 DCU control system

The control design toolbox presented in this paper serves for real-time control design and programming. The toolbox provides tools for programming, networking, and monitoring of DCU (dynamic control unit) real-time controllers. The toolbox contains most of basic process control design functionalities that can be found in many process control system programming tools. However, most advanced control design features are available as well which distinguish this design and programming tool from other similar products. The toolbox is a part of open-source and free software suit of a new advanced process control system - DCU control system [1] developed at PROSYSTEMY.

Motivation for development of a new control system arose from a frustrating experience of coming from control system research into a standard industrial and building management control environment. Advanced control design techniques are not applied here, nor control research and development tools. Smaller implementation companies and independent professionals carrying a few projects a year have an additional permanent issue - cost of corporate-build software and hardware. Progress in standard control systems (such as Siemens Simatic or Honeywell Centraline) is stalling around minor modifications in hardware electronics or existing software and updating software to newer versions of Windows. No significant innovations such as introducing advanced control system tools or applying modern information technology are visible. All that reasons led to decision to develop a new low-cost, open-source based control system. System that blends necessary industrial requirements, advanced control research tools, and that will take advantage of cutting-edge open-source and free IT platforms, as well as low-cost hardware, see Figure 1. System available to any professional individual, student, hobbyist or small company building simple or highly sophisticated control solution.

Hardware structure of the developed process control system is shown on Figure 2. It consists of real-time control segment, control system network, and visualization/monitoring/design segment. Real-time control is performed by DCU controllers (each with up 56 inputs/outputs), MODBUS RTU industrial bus and other MODBUS control devices (intelligent sensors, I/O cards,...). Networking is based on standard Ethernet/TCP/UDP/IP communication platforms. These platforms feature excellent performance, flexibility, and cost efficiency. For visualization, monitoring, process data storage, and control design any PC, miniPC, notebook, tablet, iphone may be used. Visualization (displaying actual process state) has the following mild requirements on hardware - device must have access to internet/local network and any internet browser must be installed. Remaining tasks requires additionally that the device operation system is any version of Windows, or Linux, or MacOS.

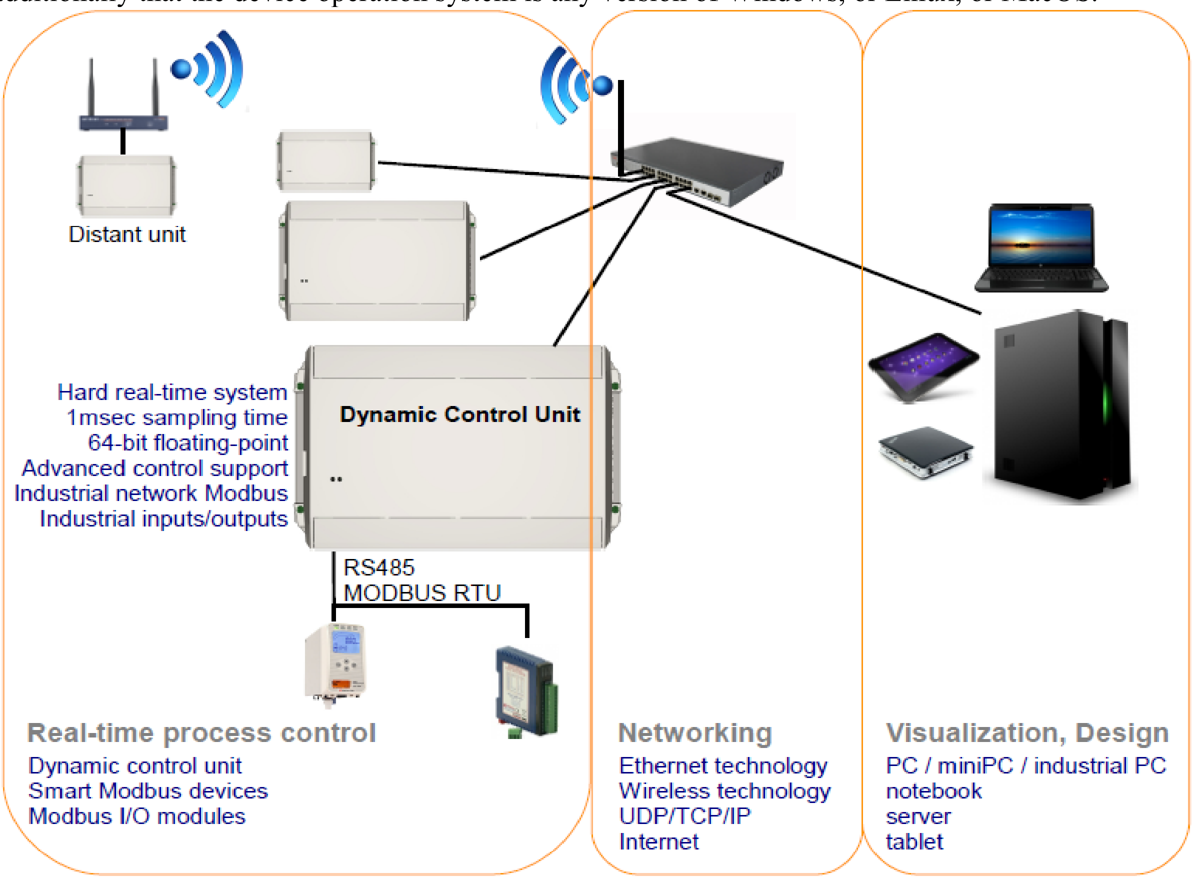

Figure 2: Hardware structure of DCU control system.

Software structure of the control system is show in Figure 3. MySQL database with UCC (User Command Center) application interface are in the heart of the software structure. Visualization is performed by web-server application Apache/PHP and custom process-related web pages. Design and programming is performed inside Matlab (or alternatively Scilab) computational platform. Inside the computational platform the DCU control design toolbox is running. It has comfortable graphical user interface and tools for programming and monitoring real-time controllers. Simulink (Xcos in case of Scilab) is used for drawing control applications (control loops, algorithms) that will be uploaded into control units.

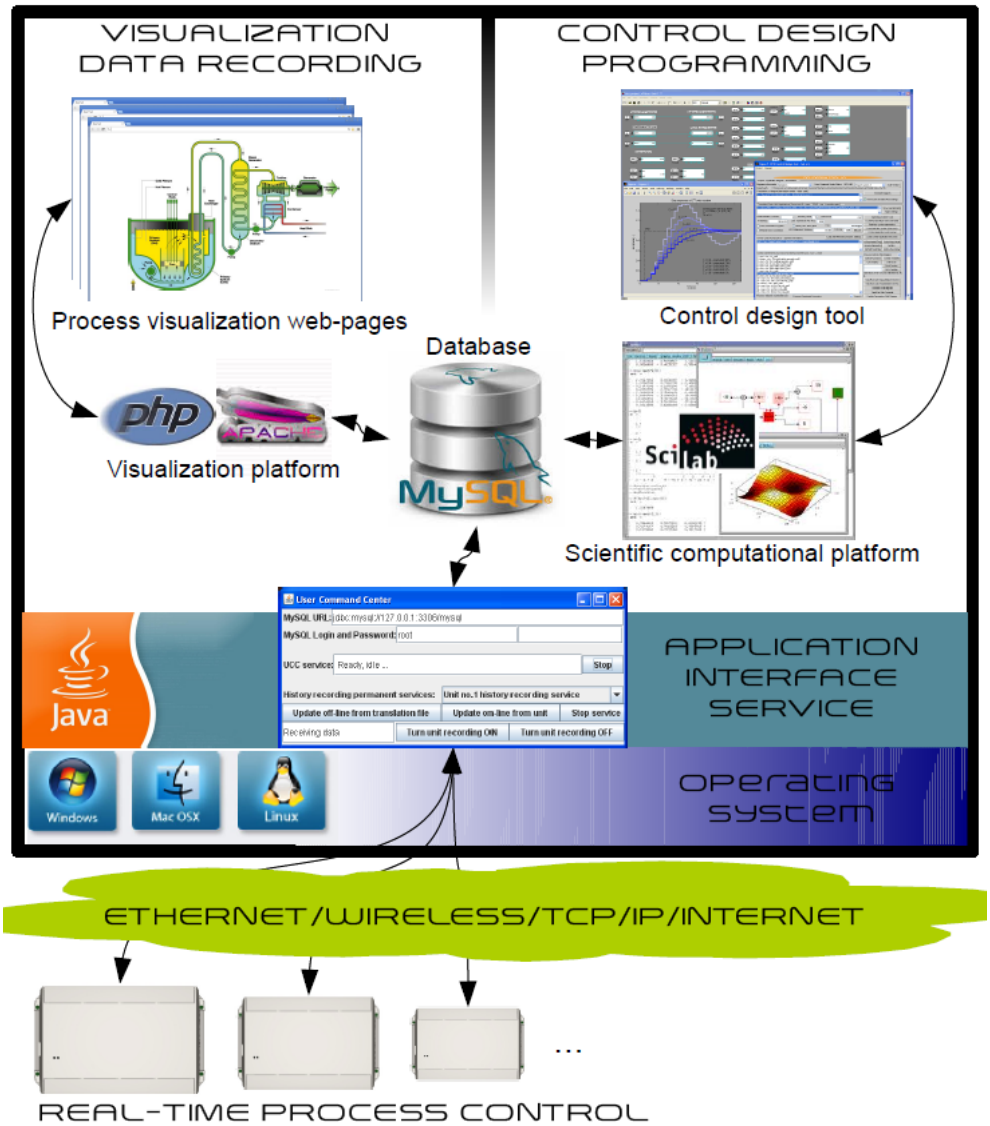

Figure 3: Software structure of DCU control system.

2 DCU design toolbox

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 4.

The first design tool is DCU design toolbox with its graphical user interface. 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.

Second tool used for control design is the Simulink toolbox. Simulink is used for programming real-time control applications. Simulink 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.

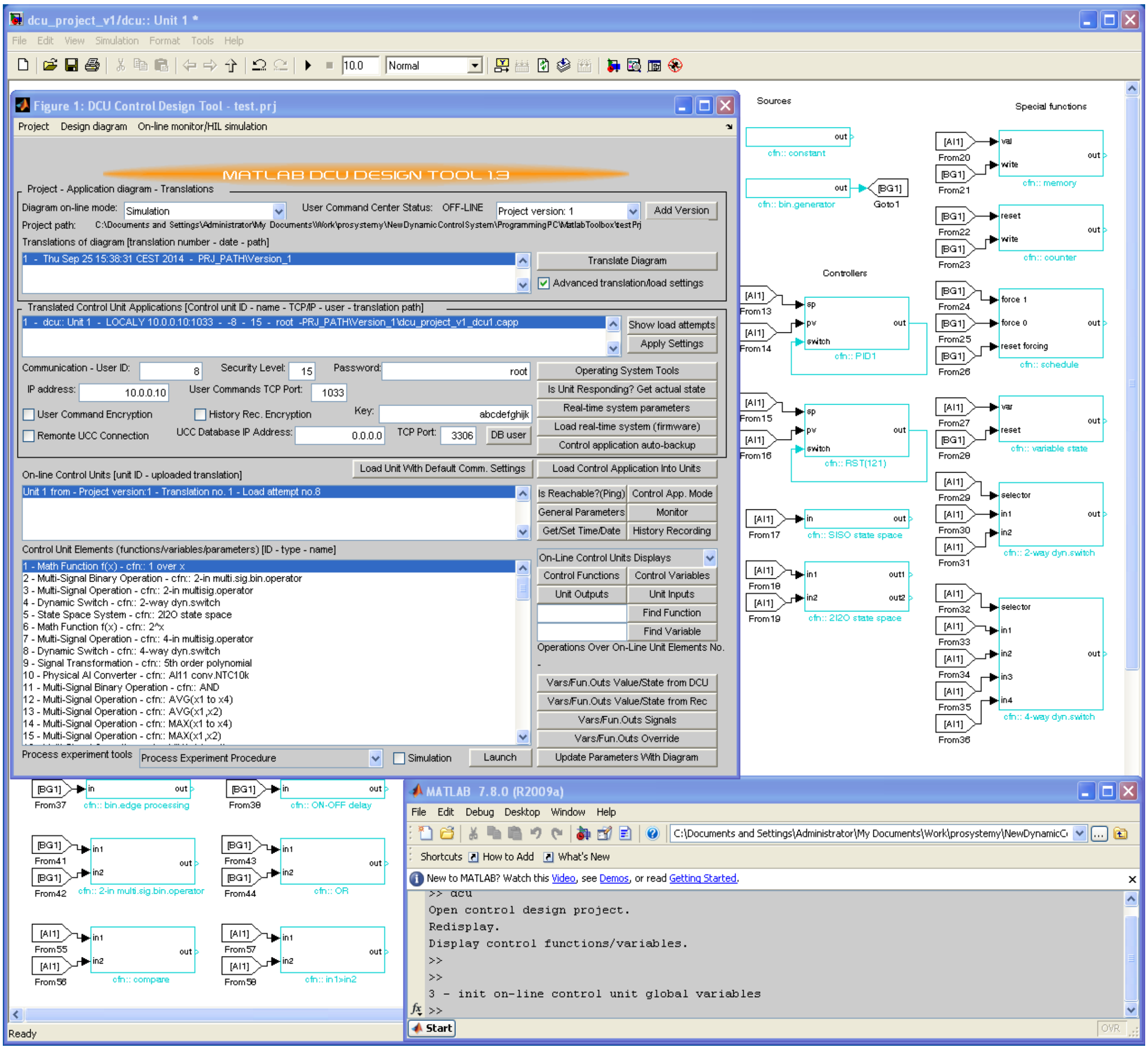

Figure 4: Graphical user interfaces used for DCU control design - DCU design toolbox interface, Simulink control application diagram interface, Matlab command line interface.

Third design tool is the main command line window of Matlab 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.

3 Step by step real-time control design

A basic step by step procedure of control design using DCU design toolbox is presented in this section. Before starting to work with the toolbox, MySQL database must be running on the designer's PC together with UCC application interface service. The toolbox is launched by typing “dcu” command. Main graphical user interface window is opened and designer must enter login and password for MySQL database.

Step 1 - Create new/open project. First, a control design project need to be created or opened inside the toolbox. In case of creating new project, user is asked for project file name and where to place it. Then it is asked to select an initial control application diagram. User may choose a diagram from a similar project or a diagram from one of examples. After selecting initial diagram, tool will automatically create Version 1.0 of the project, copy selected diagram into Version 1 folder, and open the copied diagram.

Step 2 - Programming (drawing) control application. Behavior of DCU real-time controllers is defined by control application diagram. The diagram is drawn in dynamic simulator. To open project's actual diagram go to menu Design diagram → Open actual. Simulink is started and diagram is opened. Control application for one DCU is defined inside super-block (group of blocks) which has at the beginning of its name “dcu::” string. Main control application may contain multiple DCU super-blocks. To edit control application of a DCU, double-click the DCU super-block. The actual diagram of the corresponding super-block opens and user can start add, remove, edit control functions and its connections. All control functions are in library diagram that is in DCU design tool directory. Notice, that control function has always at the beginning of its name the string “cfn::”. It indicates to translator that it is control function and not some simulation block. General parameters of a DCU (sampling times, IP address, users, Modbus settings,...) are defined in DCU superblock mask initialization.

Step 3 - Control application validation by dynamic simulation. Control application drawn in Simulink may be validated by simulating the control application diagram. Dynamic simulation requires that designer defines an operational scenario, connects a dynamic model of the process, if available, to process, place displays and graphs and launch simulation.

Step 4 - Translation of control application diagram. Validated control application diagram has to be saved and closed. It is preferable to close the diagram, to be sure that all modifications has been saved to disk. Next the button “Translate Diagram” is clicked in the main window of DCU design tool. Translation process reads diagram and save its structure into XML file ready for download (file with extension .capp). When translation done, it appears in the list of translations (first list from the top of the main window). At the end of translation, a copy of diagram with “_trsl” appended to the name is saved on disk aside with the original diagram. This copy contains ID numbers of control functions and it is to be used for on-line control monitoring and hardware-in-the-loop validation.

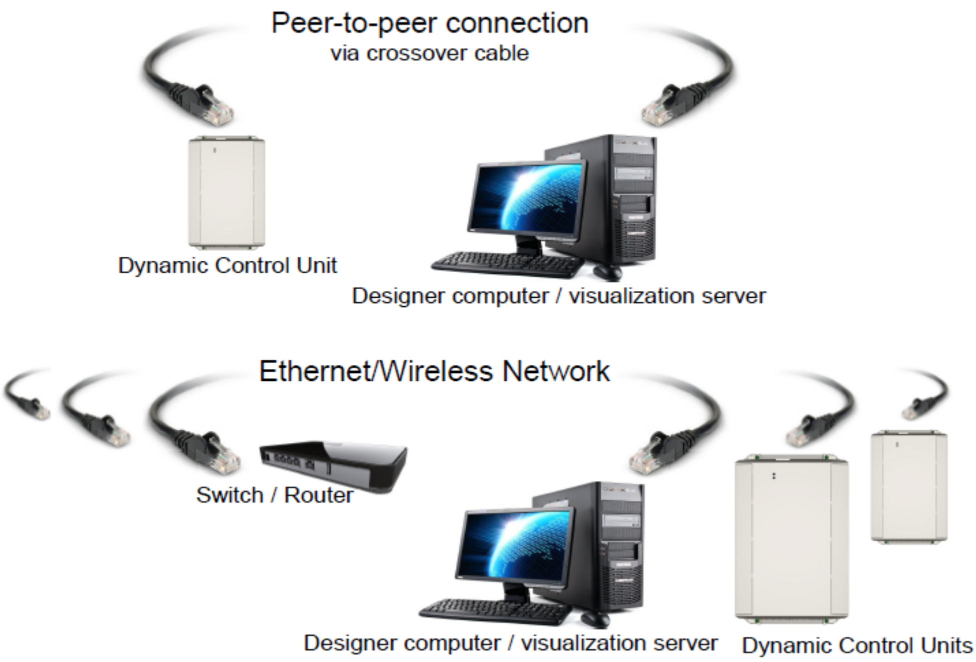

Figure 5: Connecting designer's computer to DCU control network.

Step 5 - Connect designer's computer to DCU control network. In case of connecting to one DCU, computer may be connected to DCU using Ethernet crossover cable. In case of connecting to a network of DCU (standard Ethernet local area network is used for networking DCUs), computer is connected to the network using standard Ethernet cable, see Figure 5.

Step 6 - Load translated control application diagram into DCU. By selecting a translation in the translation list, all translated DCUs are displayed in the list of translated unit applications (the list below) with its communication settings, i.e. settings for communicating with each DCU. Selecting a DCU from the list of translated unit applications will display actual communication settings of the selected DCU. These settings are initially (after translate) taken from general parameters specified for each DCU in context of the diagram. To check if the settings are correct and if the unit is rsponding properly, the buttons on the left side of the communication settings are available: “Is unit Reachable via this IP?”, and “Is Unit Responding? Get actual state”. To load translated control application into selected DCU (selected in the list of translated DCU) it is sufficient to click on the button “Load Control Application Into Units”. Communication settings are used for loading.

Step 7 - Hardware-in-the-loop (HIL) validation. In case of complex processes and control algorithms, HIL validation may be applied. It is the most advanced technique for control application validation. It validates control application that is already loaded in real-time controllers against process model simulated on designer computer. DCU design toolbox supports HIL validation. Designer needs only to switch diagram on-line mode to “On-chip simulation control” (HIL control) and open online monitor/HIL validation diagram. Next, process model has to be placed into the diagram and it has to be connected to control application. Launching simulation of online monitor/HIL validation diagram will, in fact, launch HIL simulation. Control application input values (obtained from simulating process model) are periodically sent into DCU controller. The values are processed inside DCU and computed control application outputs are sent back to simulating platform where they are injected into process model.

Step 8 - On-line monitoring of DCU controllers. To start process data recording into database the selector “Diagram on-line mode” is switched from “Simulation” to “Real-time process control”. History recording is started in DCU as well as on designer PC inside UCC service. User can now use buttons related to historical data, for example “Var./Fun.Outs Signals” allowing to display signal of a control variable. To see actual on-line values directly in diagram user may open on-line monitoring/HIL (hardware-in-the-loop) simulation diagram by selecting menu “On-line monitor/HIL simulation” → “Open on-line/HIL diagram”. This diagram is a copy of design diagram but control functions contains actual function ID numbers from translation. Buttons on the side of on-line units list allow user to check if DCU is responding, display general parameters, get/set DCU actual time and date, start/stop control application, start/stop process data recording. Buttons on the side of on-line unit elements list allows to enlist control functions/variables of selected on-line DCU (upper part) and to do operations over control functions/variables (lower part).

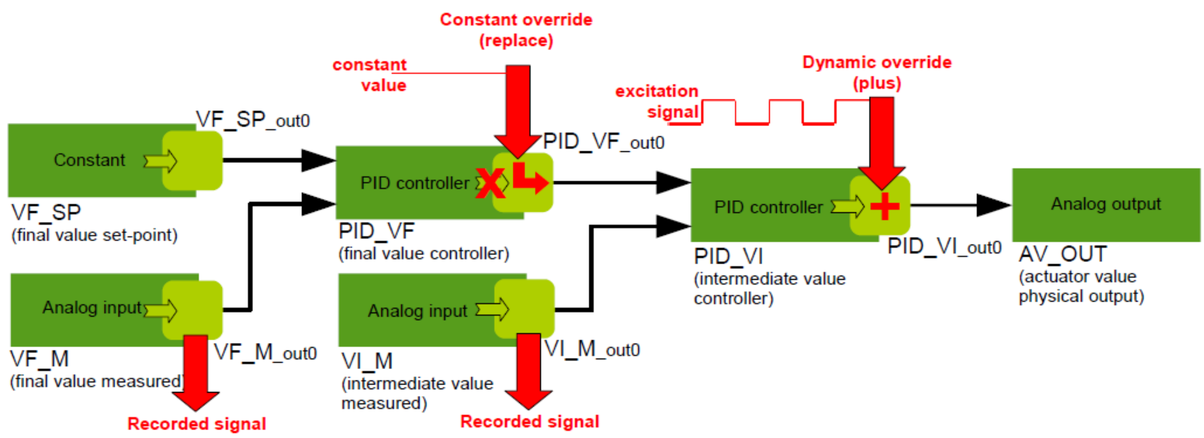

Figure 6: Example of process experiment procedure.

Step 9 - Control application tuning and optimization - process experiments. Designer may define and launch process experiment procedure which sends excitation signals into process and collects process response. A controller tuning technique may be launched as well, that uses process experiment procedure to compute optimal controller coefficients. Example of process experiment procedure used for PID controller tuning is shown in Figure 6.

4 Conclusion

DCU control design toolbox has been presented in this paper. The toolbox is a part of open-source and free software suit of DCU process control system. The role of this toolbox in the control system is to provide tools for comfortable and efficient advanced control design and programming. The toolbox satisfies not only basic needs of a process control designer, but brings features yet unseen in process control design, such as dynamic simulation, advanced control functions, process experiment tools.

References

[1] *DCU Control System: User Manual*. Available online: http://www.prosystemy.sk/downloads, PROSYSTEMY, s.r.o., Slovakia, 2014.

Hynek Prochazka
hynek.prochazka@prosystemy.sk