REAL TIME CONTROL SYSTEMS WITH THE MATLAB – SIMULINK SYSTEM USING.

Andrej Dobrovič

Abstract

In the practice there is requirement for the designing of control algorithms in the given time without detriment to quality and functionality. Thanks to some software products forwardness, new control algorithms can be designed and tested in real practice very quickly with excellent quality. The leader in this field of products is Mathworks company with Matlab system with toolboxes Real-Time Workshop and xPC Target. The tutorial of setting and using of the xPC Target toolbox is proposed in this paper.

Keywords: RT, Real-time, Matlab, controller design, developing, xPC Target.

1. INSTALLATION AND THE xPC TARGET ENVIROMENT SETTING

The installation of the xPC Target toolbox can be selected using the main installation of Matlab system or later this toolbox can be added after the main installation. When the xPC Target is installed, it is shown in the Launch Pad window in the Matlab. In the system, where Matlab is installed, there must be installed a compiler of the C++ language, for example Watcom C++, Microsoft Visual C++ on the Microsoft Windows platforms or gcc compiler on the UNIX/LINUX platforms.

Next step is xPC Target configuration to ensure the communication between Matlab and xPC Target. This can be done either by running the Enviroment Setup from Launch pad or by command typing:

xpcsetup

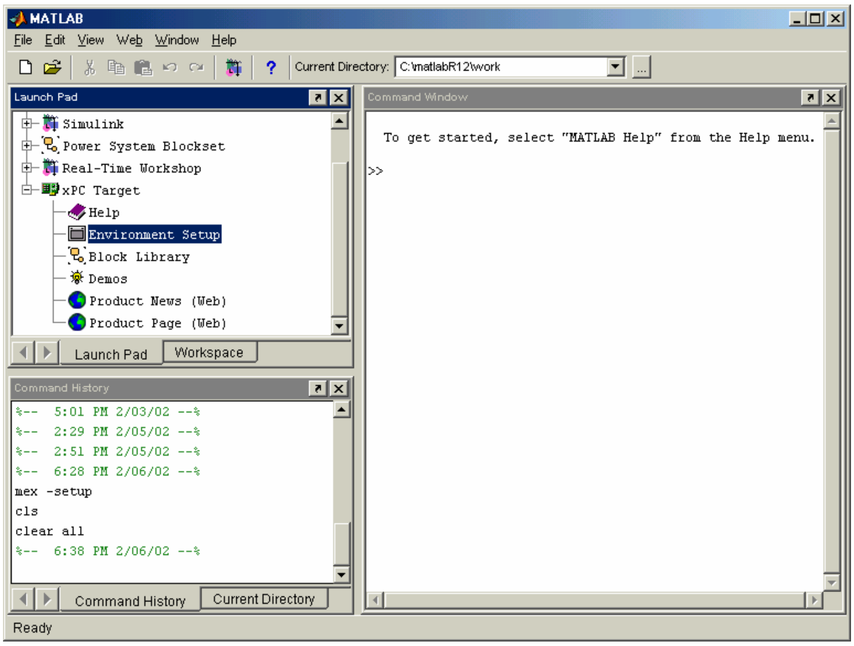

Fig.2.: Environment Setup running

After this the window is shown (figure 2) where the parameters of xPC Target must be set.

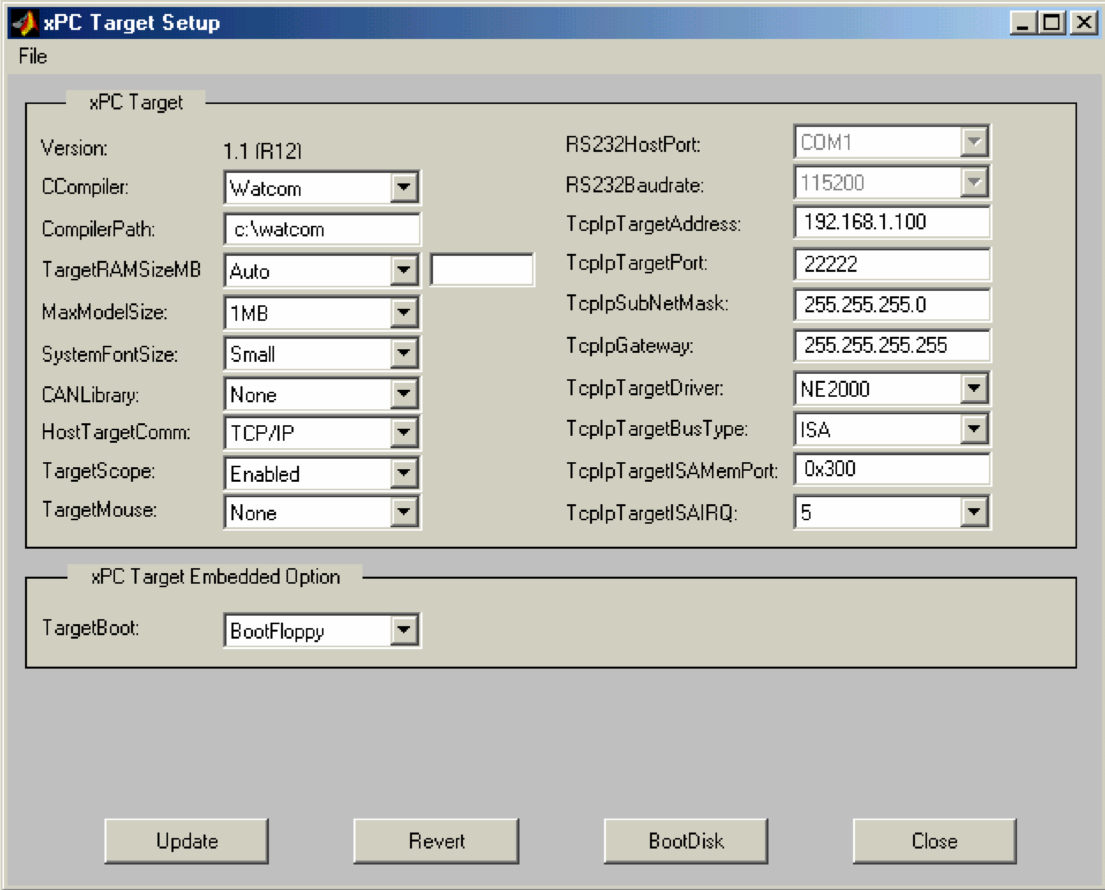

Fig. 2.: The parameters of xPC Target setting.

The parameters of the xPC Target setup will be explained now:

CCompiler

– The type of C++ used compliler – Watcom C++ or MS Visual C++.

Compiler path

The path, where C++ compiler is installed.

TargetRAMSizeMB

With the „Auto“ selected, the memory will be set automatically, in the „Manual“ mode the size of the memory must be set manually.

MaxModelSize

The size of the memory, which is used for the Simulink model.

SystemFontSize

This is the type size used on the control computer.

CANLibrary

CAN Library specification, which are used in the application

HostTargetComm

The technique of communication between control and development computer – serial communication with RS232 protocol or TCP/IP protocol on local Ethernet

TargetScope

This option will allow or deny process displaying on the control computer.

TargetMouse

This is the port, where the mouse on the control computer is connected.

RS232HostPort

Here is the specification of the communication port number on the control computer, when the HostTargetPort was set to RS232.

RS232Baudrate

It's the transfer speed between control and developing computer setting.

TcpIpTargetAddress

Here must be the control computer TCP/IP address setting. The local network can use the address in form 192.168.1.X where X is free nonused network address from range 1-255.

TcpIpTargetPort

Here is TCP/IP port of control computer. Value greater than 20000 is recommended.

TcpIpSubNetMask

TCP/IP mask of local network. With network addresses like 192.168.1.X the mask must be set to 255.255.255.0

TcpIpGateway

The specification of the gateway of network (when a local network is not connected to other network, this need not be changed)

TcpIpTargetDriver

The driver of the network card. Either NE2000 compatible card or SMC91C9X can be used.

TcpIpTargetType

It is the type of slot where the network card is connected. Choose PCI or ISA Slot.

cpIpTargetISAMemPort

The address of memory port for the ISA network card, when it is used in control computer.

TcpIpTargetISAIRQ

The interrupt of ISA network card on the control computer.

xPC Target Embedded Option – Target Boot

The disk system, for which the designed model will be written:

BootFloppy - the boot diskette creating

DOSLoader - start files creating for the control computer, when the real time operating system must be started from harddisk or flash memory

StandAlone - the control algorithm will be started in the boot time, not directly from the developing computer

After control computer parameters setting the *update* button must be pushed to set the values as actual. When the control system will be started from diskette, empty diskette must be inserted to the mechanics and after pushing the button BootDisk and question confirm Matlab will create real-time system to diskette. With this diskette the realtime control system will be started on the control computer.

2. THE QUICK EXAMPLE – SIMULATION ON THE CONTROL COMPUTER

When the diskette with the real-time system was created, it can be inserted to control computer diskette mechanics, which is connected via RS23 cable to the developing computer. Then the control computer can be switched on, and real-time system will be booted. After this we will see on the top of the control computer the next window (figure. 3).

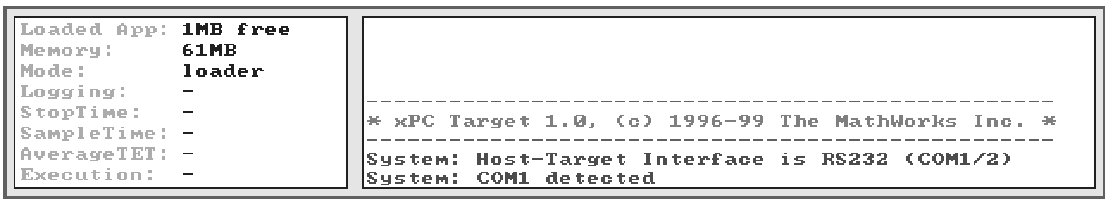

Fig. 3.: The control system startup.

The bottom of the window is empty, later there will be time responses of control and feedback values.

Now an easy test scheme in the Simulink can be created on the developing computer. For the first experiment type the next text in the Matlab command window:

xpcosc

In the Simulink there will be shown a scheme like in figure 4.

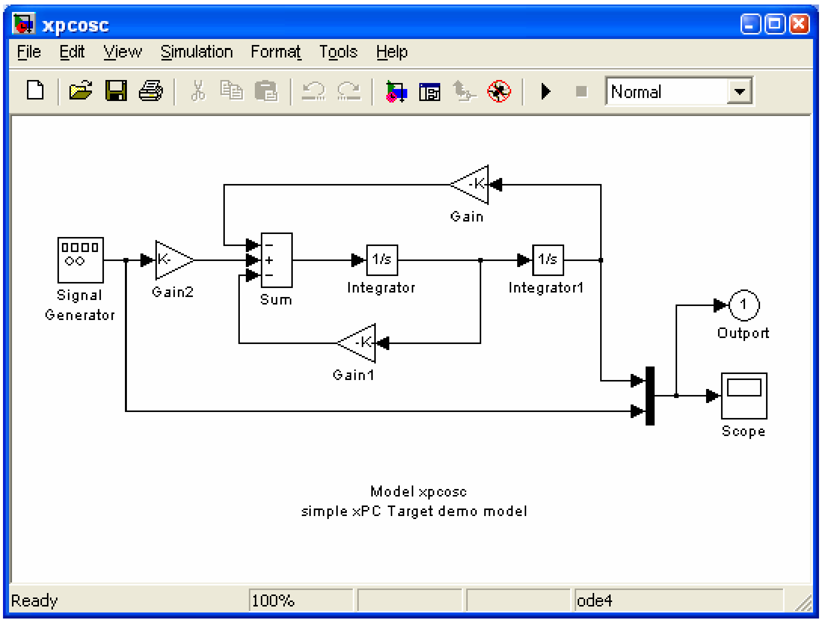

Fig. 4.: The model of the simple system

Now the simulation of the system can be started. After pushing Run button a simulation will start. In the opened Scope window an output of the simulation can be seen. When the simulation is satisfying, the model system can be prepared for control computer upload, where it will be run.

3. THE SCHEME COMPILATION IN THE SIMULINK

The first step is the setting of the target of compiling. The next option must be chosen in the menu: *Simulation -> Simulation parameters* to open the window of the simulation parameters. The fixed step of the simulation must be set. In the *Solver* it must be set *Type: fixed step* and *Fixed step size: 0,000250 s*.

Then in the *Real-Time Workshop* tab push the button *Browse*. From the menu (Figure.5) choose xPC Target and click OK. Now click the Build button.

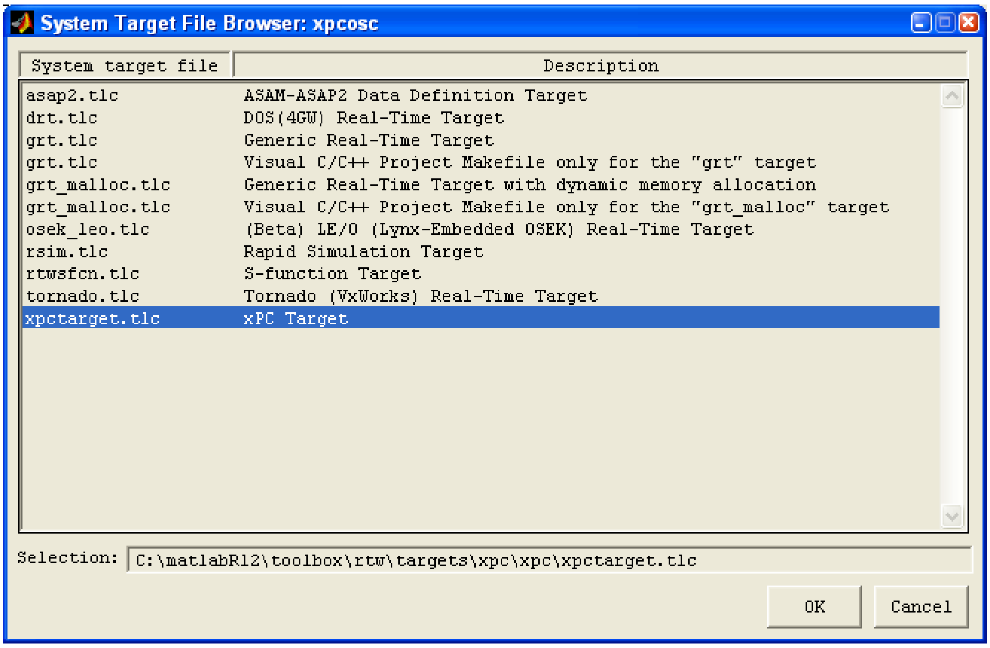

Fig. 5.: The target of compiling selection.

Now the source files will be built in the Matlab and compiler of C++ language will be used. Then the source files will be compiled and the system uploads to the remote control computer. (Figure. 6).

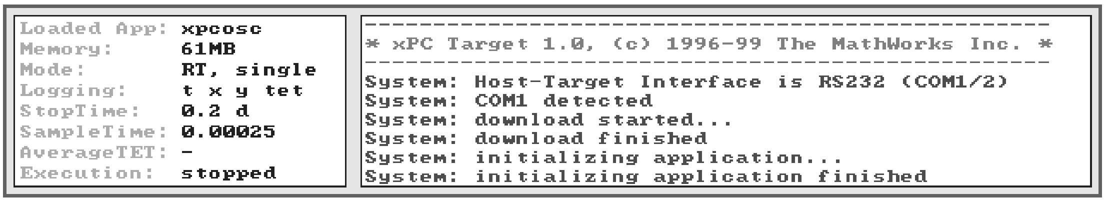

Fig. 6.: The showing of upload status on the control computer window.

4. THE SIMULATION RUN ON THE CONTROL COMPUTER

When everything to this time was without error messages, there is a real-time control system compiled in the control computer . Now in the command line in Matlab it can be written:

start(tg)

The real-time application on the control computer will run and the information about this will be shown on the top of the status window. The process will finish in 0.2 sec (the determined simulation time). The application can be started directly from the control computer, the key C must be pressed to switch to the command mode. Now it can be written:

start

and the real-time application will be run.

Each signal from the control computer can be shown in the developing computer. With the command in Matlab:

xpcscope

the window will be shown, where the signals from the control computer to be followed must be set. (Figure. 7)

Figure. 7.: The window for the osciloscope setting. (In the control computer.)

From the file menu choose *New Scope*. In the Manager window *View Scope 1* button appears. Then the new xPC Target window shows in the developing computer. With *Add/Remove* the window can be launched, where the signals, which should be followed, can be set for show. The next can be chosen:

/Integrator1

/Signal_Generator

After closing the window and pushing Start button the application can be started in Matlab command window typing *start(tg)* command. In the window *xPC Target: Scope1* actual real-time outputs of selected signals will be shown (Figure.8).

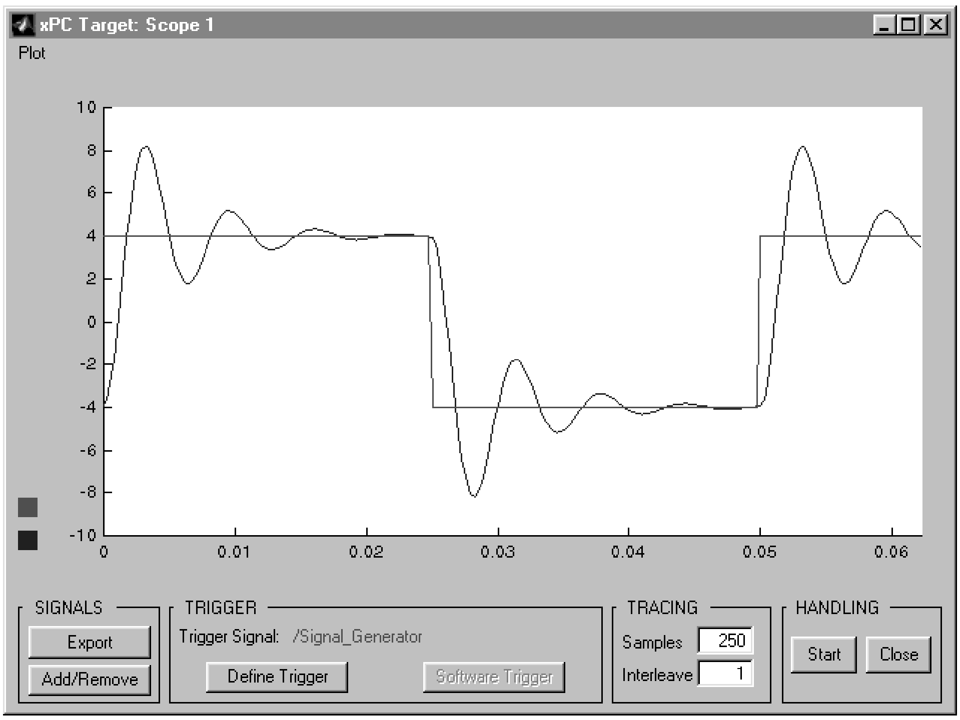

Fig. 8.: The signals from the control computer in the Matlab system viewing.

Similarly to above, the signals can be chosen here, which can be shown on the control computer display. In the Matlab command window it can be typed:

xpctgosc

and the *xPC Target: Target Manager* window will open. From the menu the New Scope option can be selected, and the empty graph will be shown. After pushing *Add/Remove* the same signals like before can be selected:

/Integrator1

/Signal_Generator

and the simulation can be run with:

start(tg)

Now the same signals in the Matlab and control computer window will be displayed. In this way any signal, which is interesting for us can be shown. Other commands for viewing and logging data from control computer are in the original documentation of the Matlab system.

5. CONCLUSION

- With the using of this type of rapid control system developing, new control algorithms can be designed and tested very quickly. Because the control computer uses the system of real-time, there is no problem for testing control algorithms with sampling period about 10 μs. The real-time control computer can be classical PC, industry computer, special DSP controller or many others. This control system can be connected with analog and digital boards via senzors and actuators to real system such as AC motor. With this rapid system developing, the time from the designing of a new control system to using the new system in practice is reduced to minimum.

LITERATURE

Original Mathworks system help for Matlab and Simulink system

Author:

Ing. Andrej Dobrovič, KASR FEI STU, Bratislava

dobrovic@kasr.elf.stuba.sk