

Application Note

AN2395/D
12/2002

PC Master
Software Usage

By Milan Brejl and Pavel Kania
S³L Applications Engineerings
MCSL Roznov pod Radhostem



Introduction

The PC master software is a PC Windows®-based application providing various kinds of support for embedded applications. It is a very efficient tool that can be used for debugging, monitoring, and controlling the target application and for demonstrating the board application functionality.

The embedded application target board is connected with the PC via a standard RS-232 serial line. The communication protocol on the base level enables the user to:

- Read/write board application variables
- Record or track variable movements
- Send application commands with parameters

The PC master software takes advantage of these base-level functions and enables the user to:

- Watch board application variables — either in the original format, or transformed into a more easily viewable format
- Modify variable values (in the original or transformed format), and in this way, control the board application
- Plot the course of read variable value movements using the Scope feature
- Plot the course of fast events using the Recorder feature
- Control the target application using application commands
- Add HTML pages providing help, or descriptions for the target board application features, scopes and recorders, application control, etc.
- Create a logical tree structure of blocks relating to the various board application functions, and assign their corresponding description pages, scope and recorder definitions, and watch grids
- Add an HTML based Control page that can control the board application, using graphically attractive web standard tool

Windows is a registered trademarks of Microsoft Corporation in the U.S. and/or other countries.

Application Window Description

The PC master software application window is shown in **Figure 1**. The window consists of various panes: the Project Tree pane, the Detail View pane, and the Watch-grid pane.

The **Project Tree pane** shows the contents of a PC master software project. Users can add project sub-blocks, Scope, and Recorder definitions to the project block, in a logical structure to form a Project Tree. This pane provides point and click selection of defined elements.

The **Detail View pane** dynamically changes its content depending on the item selected in the Project Tree. This pane provides tabs to pages containing additional information associated with the selected item. For example, the control page, the algorithm block description page, or even the scope or recorder chart page, with an associated current item help page.

The **Watch-grid pane** contains the list of watch variables. You can monitor these variables as well as changing their values (if enabled in the variable definition).

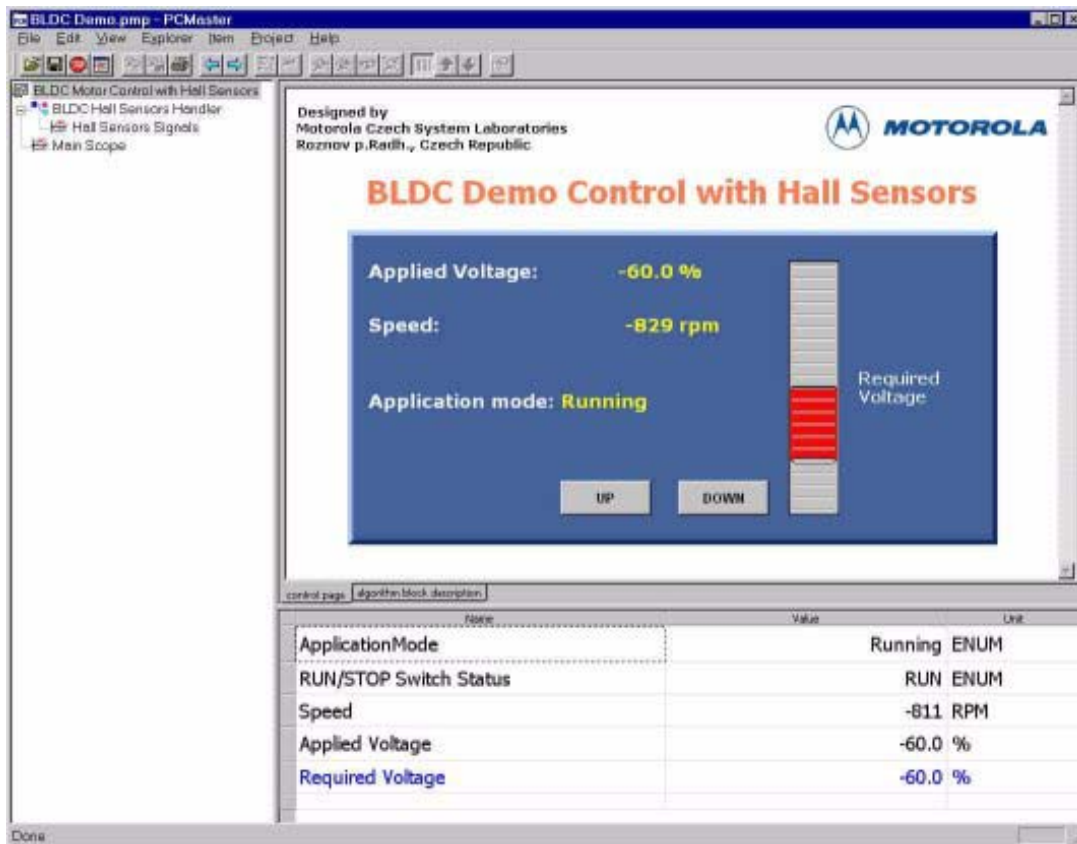


Figure 1. PC Master Software Application Window

All of the information related to one project is stored in a single PC master software project file with the extension “.pmp”. This information includes the project settings and options, the Project Tree, Detail View pages, real-time chart definitions, watch interface settings, variables and commands, stimulators, and other details.

PC Master Software Variables

One of the most important ways to debug the embedded application is to read the target application’s variables and write to them. These simple actions give the user the possibility to fully control any application. To start working with PC master software variables, their names and addresses have to be defined. In order to make this step easier, PC master software supports various types of map files (e.g., CodeWarrior elf and HiWare map formats).

Once the variables and their addresses are loaded or manually defined, the user is able to choose the format in which the content will be interpreted (decimal, hexadecimal, binary, and ASCII formats are supported). If the variable represents a linear quantity (e.g., voltage), the content can be transformed to the correct value using linear transformation available in the PC master software application.

In **Figure 2**, you can see an example of the PC master software variable definition with linear transformation to obtain the correct value of the rotor’s revolutions. In **Figure 1** you can see the Speed variable in the Watch-grid pane. Here, the variable represents the rotor’s revolutions in RPM. The minimum value of the 16-bit variable (0) corresponds to 0 RPM, the maximum value (32767) corresponds to 1500 RPM. The coordinates of the first point are [0, 0] and the coordinates of the second point of the linear transformation are [32767, 1500].

This simple example only demonstrated one unique feature. Many other functions can be defined to effectively display variable information.

Embedded applications usually contain many status variables whose contents indicate the current state of the application. Wouldn’t it be great to see a text description of each state, instead of an impersonal number? That can also be done with PC master software with the possibility of enumerating any variable and assigning several text values to it. As an example, **Figure 3** shows the configuration of the ApplicationMode variable. This variable was highlighted in **Figure 1**.

To further the list of PC master software variable features, we should not forget the capability of monitoring minimum and maximum variable values, mask definition (used to write variable content), defining refresh rate of each variable, etc.

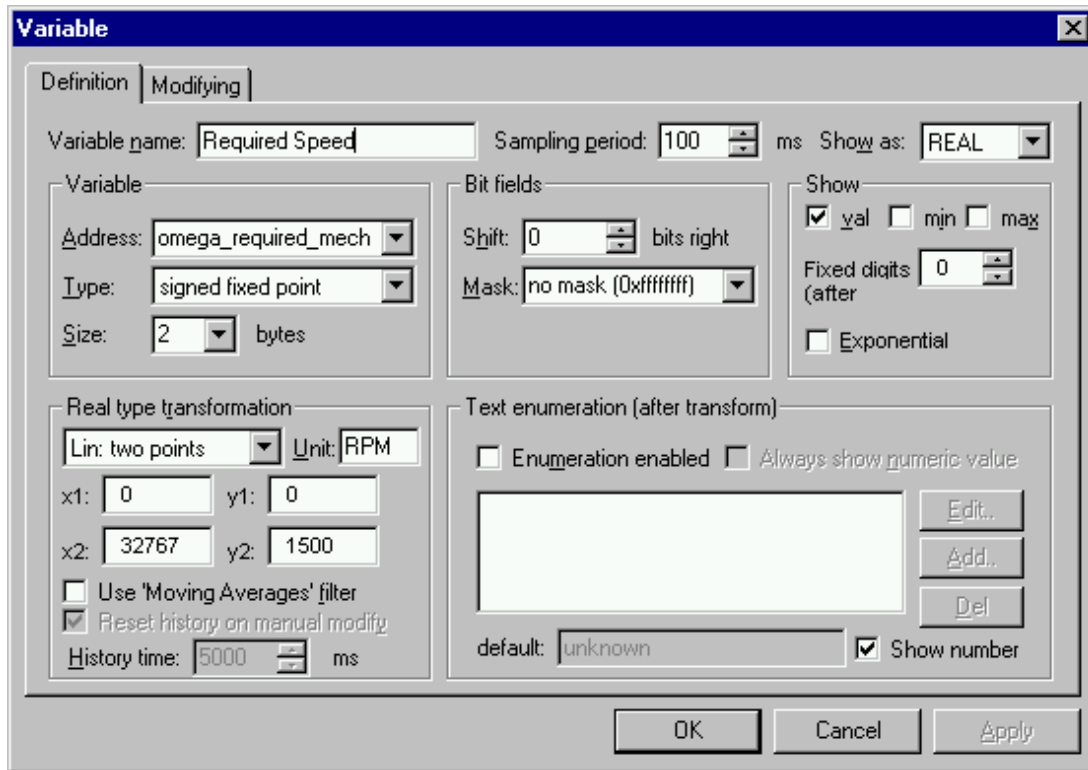


Figure 2. Example of Real-Type Transformation of a PC Master Software Variable to Obtain the Correct Quantity Value

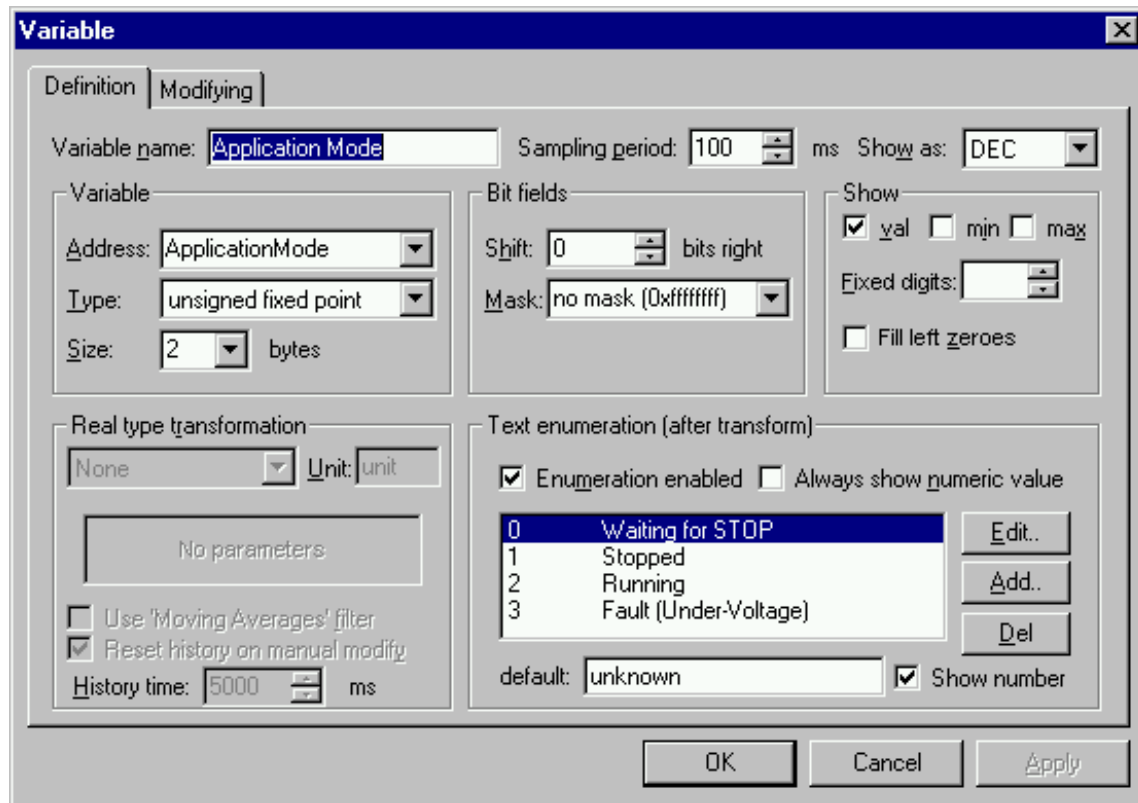


Figure 3. Displaying the Status Variables with Enumeration

PC Master Software Scope

PC master software Scope has highly optimized commands for minimal data flow on the line. It offers the user the choice of reading one to eight variables in each 10 ms or less period. This feature can be typically used to chart variable movements in real time. **Figure 4** is an example of a PC master software scope display, showing signals from the Hall sensors in a brushless dc motor (BLDC).

The Scope display may be configured in many ways. For example, variable movements can be displayed together or spread over several graphs.

Figure 5 shows the Scope configuration window which generates the data for **Figure 4**. Hall sensor variables are checked so that they can be tracked by the scope and they are assigned to one graphic block. Variable movements will be displayed in one chart with one Y axis (see **Figure 4**).

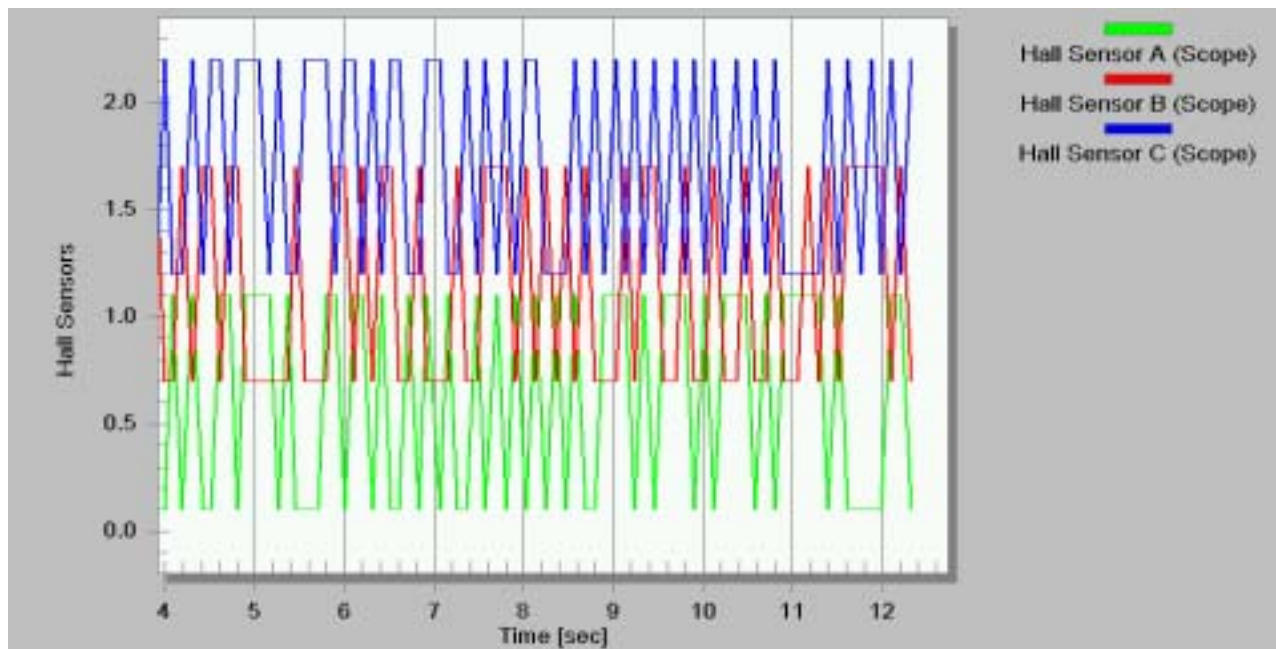


Figure 4. PC Master Software Scope Display with Signals from BLDC Hall Sensors

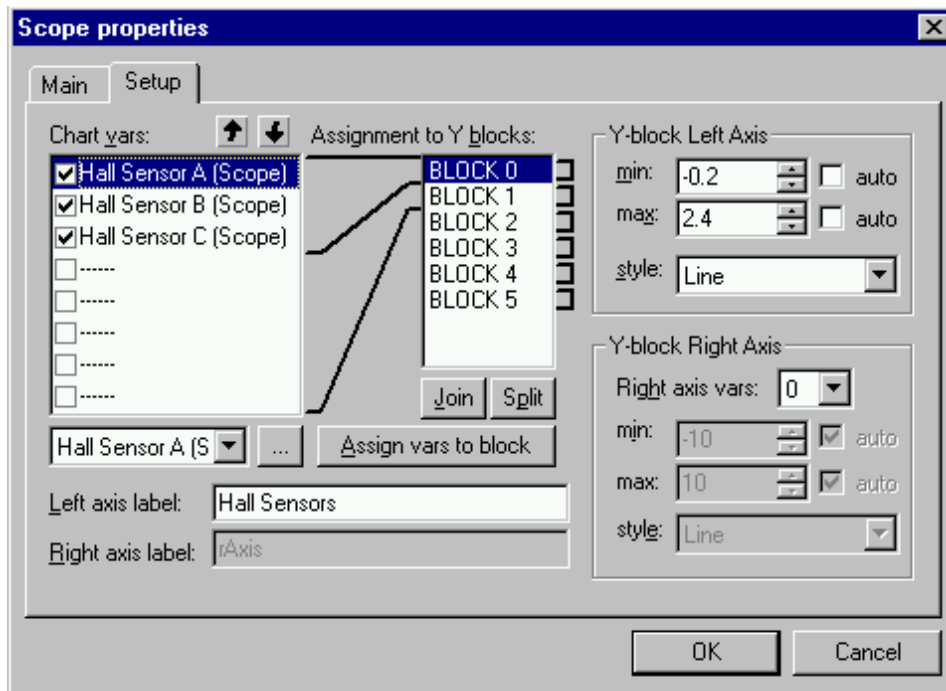


Figure 5. PC Master Software Scope Configuration Window

PC Master Software Recorder

The PC master software Recorder feature is dedicated to watching those fast changing variable movements that cannot be shown on-line. The recorder is implemented as a stand-alone routine in the target application code. This routine controls the sampling in the internal buffer. The Recorder sampling period is equal to the period required to launch the Recorder routine in the target application code, and controls the sampling of the internal buffer. Sampling begins as soon as the recording routine is called; therefore, the launching period may be in the range of microseconds.

The PC master software Recorder may be configured in many ways, with its strongest feature being the various definable trigger conditions. In the configuration window the user can choose positive, negative, or both slopes, and threshold values for the trigger. The number of pre-trigger samples and the number of total samples which can be stored in the Recorder buffer, are sufficient to control monitored movements.

Generally speaking, the Recorder should be used when the Scope is no longer sufficient because of the high sampling rate or the high amount of data being transferred through the RS-232 line. **Figure 6** is a typical Scope chart which can be obtained when the signal is undersampled. The Scope was configured to seek new data every 50 ms. Because the saw signal periods are relative to the sampling period, the acquired data does not correspond with the actual real shape of the saw signals.

The only solution in this case is to use the Recorder to acquire the correct data, shown in **Figure 7**. The sampling frequency is equal to the frequency of calls of the Recorder routine (i.e., 1 ms).

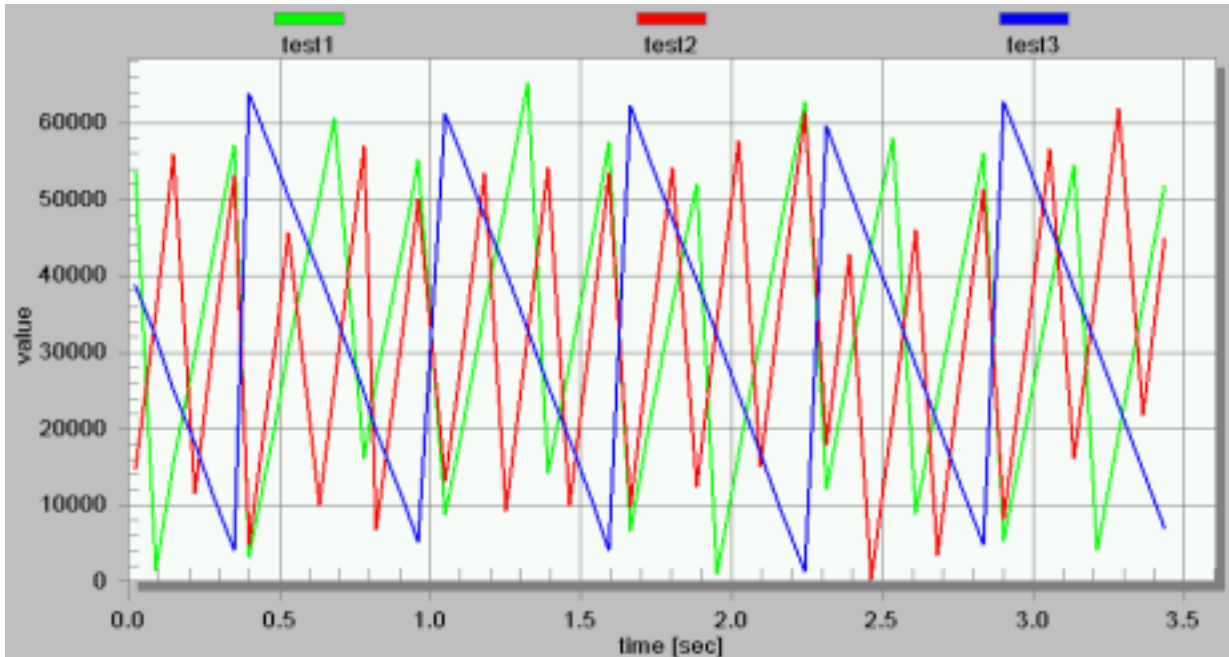


Figure 6. PC Master Software Scope Chart of the saw Signals with Sampling Period of 50 ms

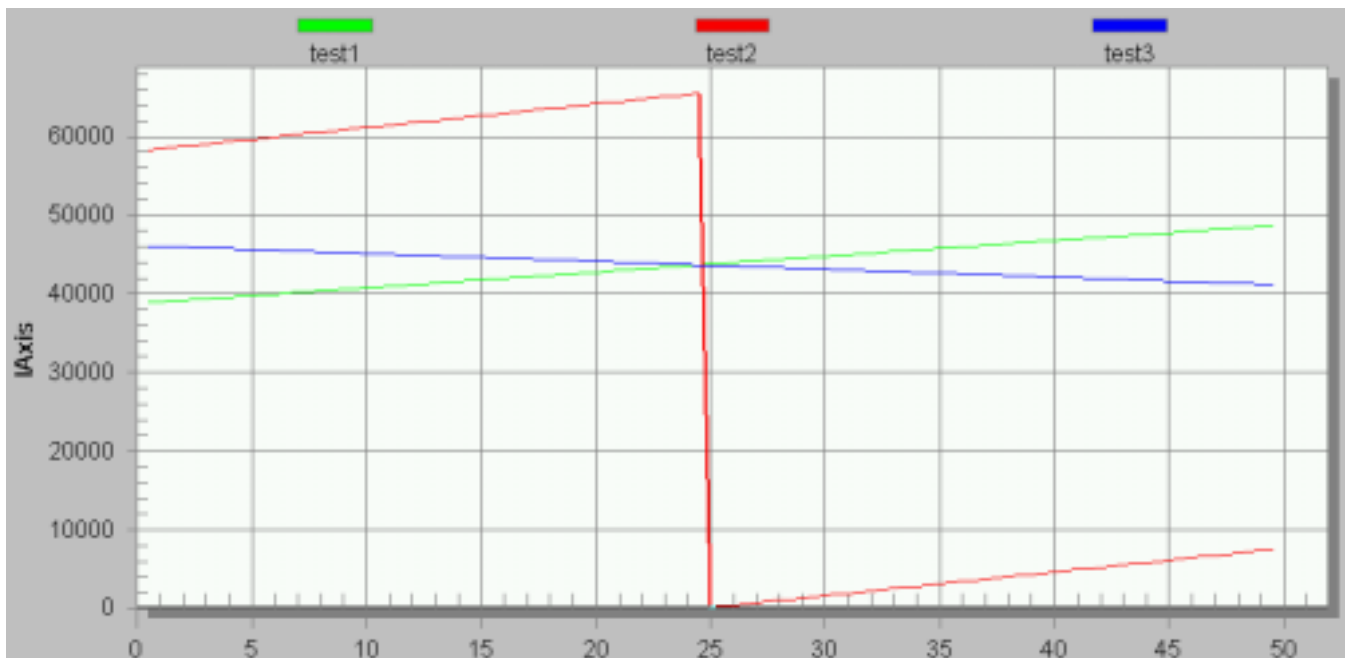


Figure 7. PC Master Software Recorder Chart of the saw Signals with Sampling Period of 1 ms

The Difference Between Scope and Recorder

The Recorder provides monitoring/visualization of application variables that are changing at a faster rate than the oscilloscope sampling. While the Scope periodically reads PC master software variable values and plots them in real-time, the Recorder is running on the target board. Variable values are sampled into and on the board buffer, then the data is downloaded from the board to the PC master software Recorder window. This mechanism allows a much shorter sampling period, which enables the sampling and plotting of very quick movements.

Variety of PC Master Software Usage

PC master software is a very efficient tool, consisting of all the base principles that allow a great variety of uses. Simply, PC master software can be used anywhere where communication with the board application brings some advantages. Let us propose some of the uses of the PC master software.

The PC master software is a great tool for **debugging**. The developer can very easily select and arrange all important variables in the Watch-grid Pane and check or update their values. Further more, the developer can track the variable movements using Scopes. This is very useful in cases where a variable's value changes instantaneously and is too fast to be read. For even faster events, like transient phenomenons, the developer can record the precise course of the variable using the Recorder, and then explore the recorded movements on a chart. Both Scope and Recorder charts can be copied to be reused in documentation.

For board applications where development is finished, the PC master software can be used for **monitoring** an application run. Here the project tree can be of great benefit. The blocks of the Project Tree Pane can create a logical tree structure, corresponding to the application functional elements. A different watch-grid can be defined for each block, as can the description HTML page Scopes or Recorders. The Watch-grid Pane is sensitive to the block selected in the project tree, which means that only those values and variable movements relevant to the selected application function are monitored.

Monitoring is a passive activity between the PC master software site and the board application. An active activity, that of **control**, is also possible. The easiest way to control board applications, is a manual modification of the variables in the Watch-grid. The control variables must not be read-only. But the PC master software has a special feature for control — the *Control Page*. The Control Page is one of the pages in the Detail View pane. The tab to this page is not sensitive to the block selected in the Project Tree, and is always available. The Control Page is a special HTML page. By using all of the HTML language web standards, it can be used to create a graphically attractive control center (see [Figure 8](#)).

3-Phase AC Ind. Drive with V/Hz Algorithm - Open Loop

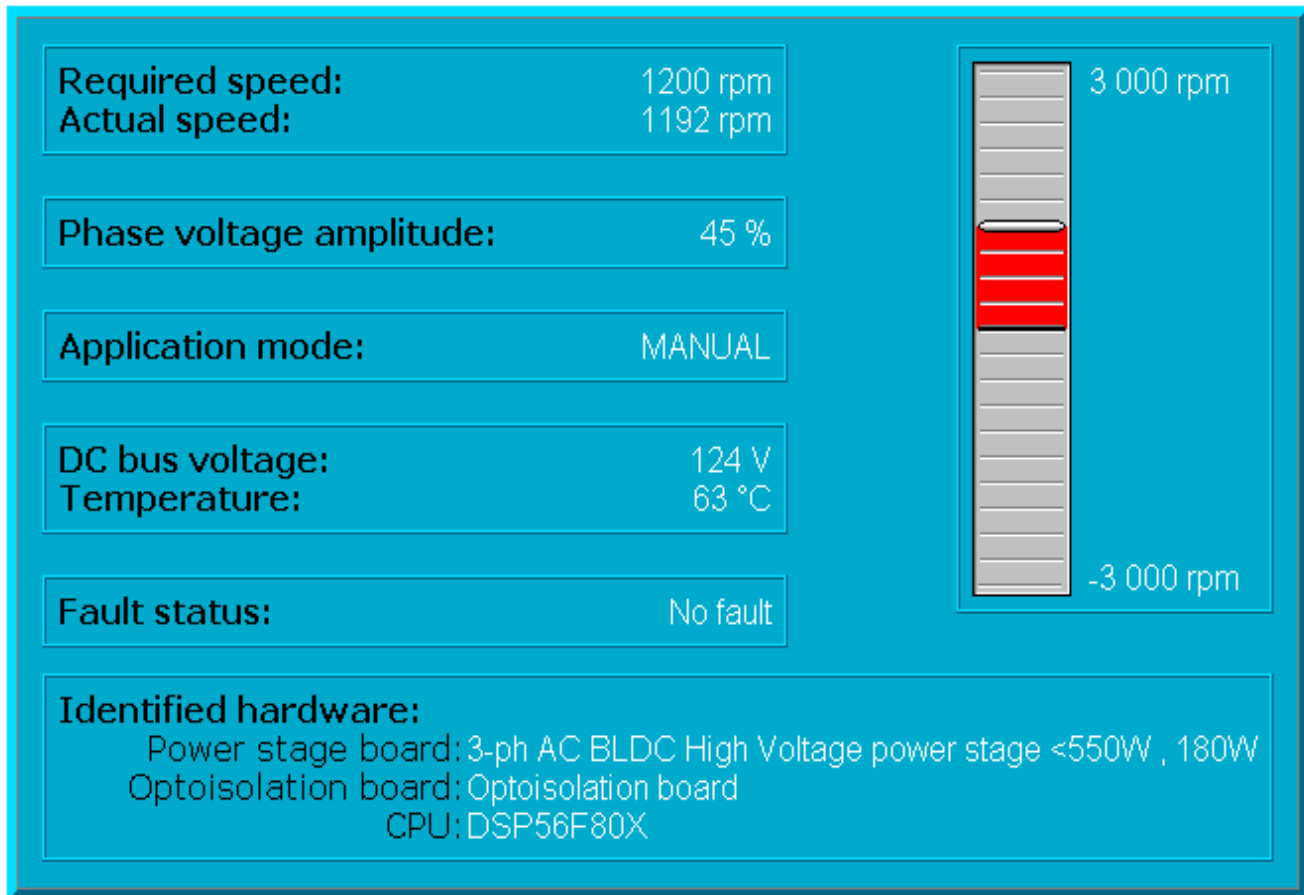


Figure 8. HTML Control Panel on Control Page

The control functionality of the Control Page is enabled by an ActiveX component provided by the PC master software. The HTML Control Page script can call functions of the ActiveX component such as:

- ReadVariable (reads from a PC master software-defined variable)
- WriteVariable (writes to a PC master software-defined variable)
- SendCommand (sends a PC master software-defined command)
- Other commands (e.g., for recorder manipulation)

The HTML script language can be VBScript or Java™Script. The creation of advanced Control Pages will be described in a separate Application Note.

Java and Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States and other countries.

PC master software enables the motor control board application to be monitored or fully controlled from a PC. A PC master software project may include description and help pages for each project tree block. Such an application may be presented in a show, or during a training session, to demonstrate its functionality. The PC master software running this project can itself become a fully featured demo-application, by simply switching PC master software to the password protected demo-mode. At a show, the public can try out all of the board's application functionality, but can not change any of the PC master software settings, variables, scope, or recorder definitions that would ruin the demo.

Conclusion

The PC master software is an Windows based application providing various kinds of support for embedded applications. Reading and setting PC master software variables, scopes, and recorders makes the PC master software a powerful tool which is ready to effectively demonstrate a board's application features. The graphical user interface makes available both:

- A range of tools for real-time debugging of any piece of user software
- The presentation of a safe, tamper-proof, user application demonstration

These attributes make it widely suitable for those developers who need to effectively speed up their development process, and to easily demonstrate the behavior and characteristics of their finished applications.

How to Reach Us:**Home Page:**

www.freescale.com

E-mail:

support@freescale.com

USA/Europe or Locations Not Listed:

Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:

Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:

Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:

Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:

Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

