A Graphical User Interface for a Dynamic Signal Analyzer Using Simulink

Meader Woo, John M. Watkins
Department of Electrical and Computer Engineering
Wichita State University
1845 Fairmount
Wichita, KS 67260-0044
J.Watkins@IEEE.org

Introduction

The ability to understand and utilize the frequency response of a linear system is a critical building block in many undergraduate engineering disciplines. For example, undergraduate students in electrical engineering will often see the frequency response in courses on circuits, signals and systems, communications, and control systems. Yet despite this significant exposure to and need for the frequency response, many undergraduate students have limited intuition on what it means and represents. As we have learned in other areas of engineering education, for students to understand a concept they need to work with it, either in simulation, or ideally in experimentation. Unfortunately, equipment for measuring the frequency response, such as a dynamic signal analyzer or frequency response analyzer, is often too expensive and complex for a typical undergraduate engineering lab. Furthermore, it does not allow to students to experiment in simulation, when an experimental lab is either not feasible or desired.

Consequently, in Watkins[1], we developed a virtual dynamic signal analyzer in SIMULINK for measuring the frequency response. In this paper, we discuss a Graphical User Interface (GUI) that has been developed to make the virtual DSA intuitive and easy to use for the students. To utilize the dynamic signal analyzer, the user places a dynamic signal analyzer (DSA) block in their SIMULINK model file. The DSA block has two inputs: the input and output signals from the linear system. The DSA block has one output, a sinusoidal signal that is swept through a series of frequencies. By measuring the frequency response at discrete frequencies, it provides a frequency response measurement with a high signal-to-noise ratio. More importantly, it allows the students to see the input and output signals at the different frequencies as they are being measured. The input and output signals in the time domain and the magnitude and phase of the frequency response are plotted in real time.

To run the virtual DSA in simulation mode requires only SIMULINK. However, if combined with the Quanser WinCon software and hardware input/output board, it can be used for measuring the frequency response of experimental apparatus in the laboratory. If the linear system is mechanical, the students can compare the system's movement with the responses they are seeing on the screen.

Other authors have used SIMULINK to create a virtual DSA. Wang, Abramovitch, and Franklin created a virtual DSA for verifying measurements and models of linear and nonlinear systems. The virtual DSA they developed was for simulation purposes only and had no interface with real hardware.[2]

On the other hand, the DSA developed by Lilienkamp and Trumper was written specifically to work with the dSPACE controller board. Besides the analog to digital (A/D) and digital to analog (D/A) converters, which you also find on the Quanser input/output board, the dSPACE controller boards have an on board digital signal processor (DSP). The onboard DSP provides for faster processing, but at a higher cost that Quanser input/output board.[3]

This paper will begin with an overview of the Dynamical Signal Analyzer Graphical User Interface (DSAGUI). This will be followed by a discussion of how the DSAGUI can be used in simulation. The details of how the DSA works will be discussed from a software and mathematical perspective. The paper will conclude with an example that demonstrates how the DSAGUI can be used in the laboratory to measure the frequency response of experimental apparatus. The DSA is available from the author upon request.

Dynamic Signal Analyzer Graphical User Interface

This GUI is based on a virtual DSA function discussed in Watkins[1] that works with SIMULINK to measure the frequency response of a user defined system. SIMULINK allowed the DSA tool to be represented as a block with two inputs, the input and output of the system to be measured, and a single output used to excite a plant with a sinusoidal signal.

After setting up the SIMULINK model, one only has to double click on the DSA block to open the DSAGUI as seen in Figure 1. From there, the user must input values into the argument boxes which include: ω_i and ω_f, the initial and final frequencies in (rads/sec) that the DSA will sweep for the frequency response, n, the number of frequencies at which the DSA calculates the frequency response, and t_s, the estimated settling time of a plant in seconds. If the sampling period at which the experiment or simulation is run, T_s, is entered the program will correct for the phase loss due to sampling delay. The user can also enter the gain and offset of the sinusoidal output signal of the DSA block.

Hitting the green play button located near the bottom center of the GUI starts the DSA function. While the DSA is running, the process can be paused or aborted if so desired. When finished the DSAGUI restores functionality to the GUI allowing the user to change any of the parameters again to run another analysis. Also integrated into the GUI are options to save the final frequency response of a successful analysis to a variable in the workspace or to a file via the export functions located in the file pull down menu. The frequency response export saves the results in a Frequency Response Data Model format which requires the MATLAB Control Toolbox. The GUI's parameters, which includes the all the DSA argument values and system model, can be saved for reference or later use.

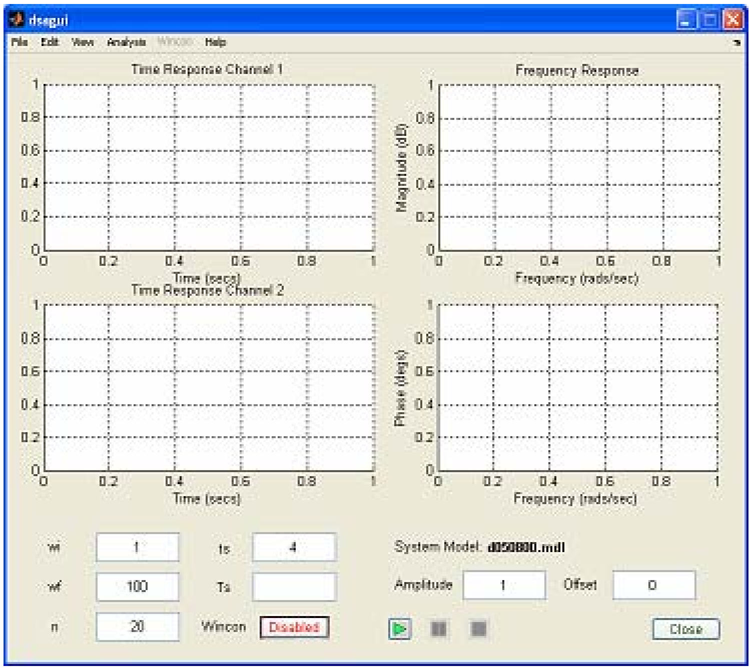

Figure 1: DSAGUI when first opened.

Using the DSAGUI with Simulations

One of the ways the DSAGUI can be used is by creating a Simulink model to run the DSA in pure simulation. This can be very advantageous for a classroom setting where using simulations also allows students to be exposed to a variety of systems without the cost of having lab equipment for each student.

For example, we could use it to find the frequency response of an unstable sampled-data system in order to test and refine a digital controller. The continuous time plant is given by

$$G(s) = \frac{100}{(s+1)(s-1)}.$$

The system is sampled at 0.01 seconds. Because the system is unstable, an initial controller is needed to stabilize the system. The following PID controller stabilizes the system.

$$D(z) = \frac{12.1187(z-0.99)(z-0.9613)}{z(z-1)}$$

Now using Simulink we can connect this system to the DSA block as shown in Figure 2.

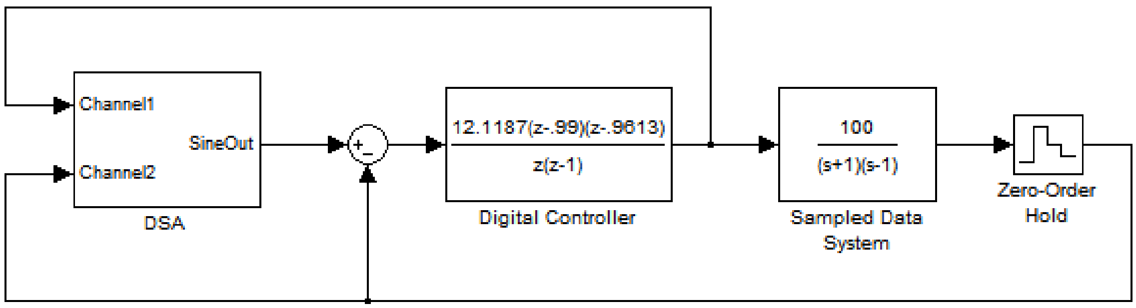

Figure 2: Block connections in SIMULINK for simulation example.

The DSA was then configured to measure the frequency response at 20 different frequencies from 0.1 to 300 (rads/sec). The settling time was set to the measured value of 0.703 seconds. Because this is all in simulation and there is not an actual physical plant, the Wincon button remains toggled off. The amplitude and offset values will remain at their default values of 1 and 0. Besides setting the channel display to "merged" or "individual," all the user must do is hit the play button to start the DSA. Once started the DSA begins to display the channel time responses and the frequency response. The original DSA displayed the time response in a single axis, however in some cases the amplitude of one channel may obscure (Figure 3) the other and a change of view from merged to individual may be used (Figure 4). With the Time Response window/s, the user is able to watch the two channel signals change in frequency and allows the user to see the amplitude and phase differences that MATLAB's Bode command does not.

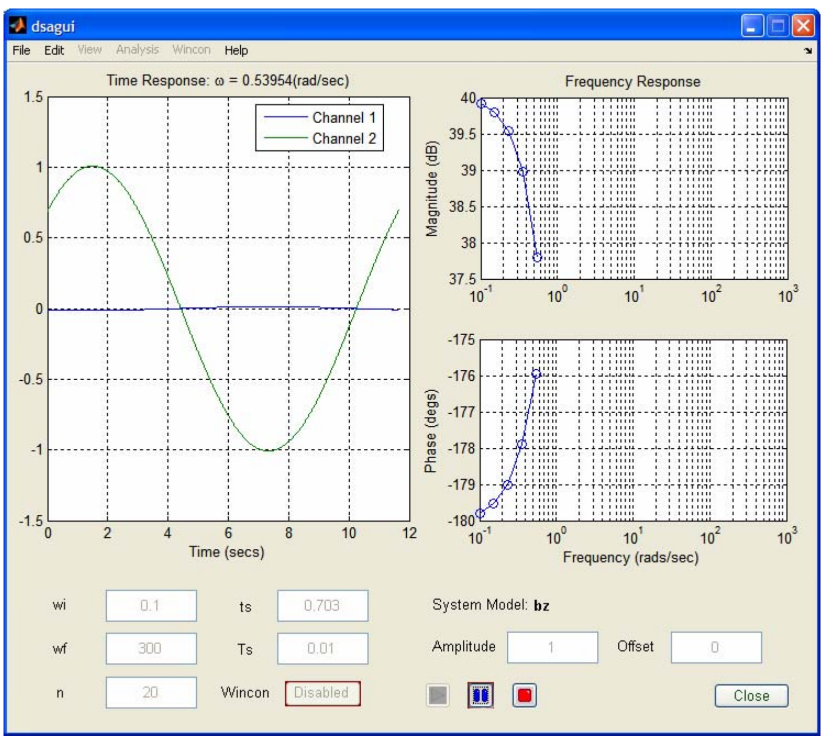

Figure 3: DSA in middle of analysis with Merged Channel View.

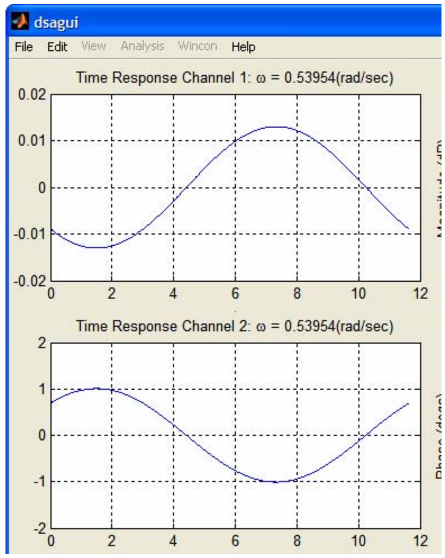

Figure 4: DSA in middle of analysis with Individual Channel View.

After the DSA finishes, a closer look at the frequency response can be achieved using the Analysis/Bode menu item. The DSAGUI can also export the frequency response data to the workspace or a file using the Frequency Response Data model format. Using this feature a comparison of the DSA's results and a pure mathematical, theoretical, frequency response of the system can be made (Figure 5). The frequency response can then be used to refine the digital controller as needed.

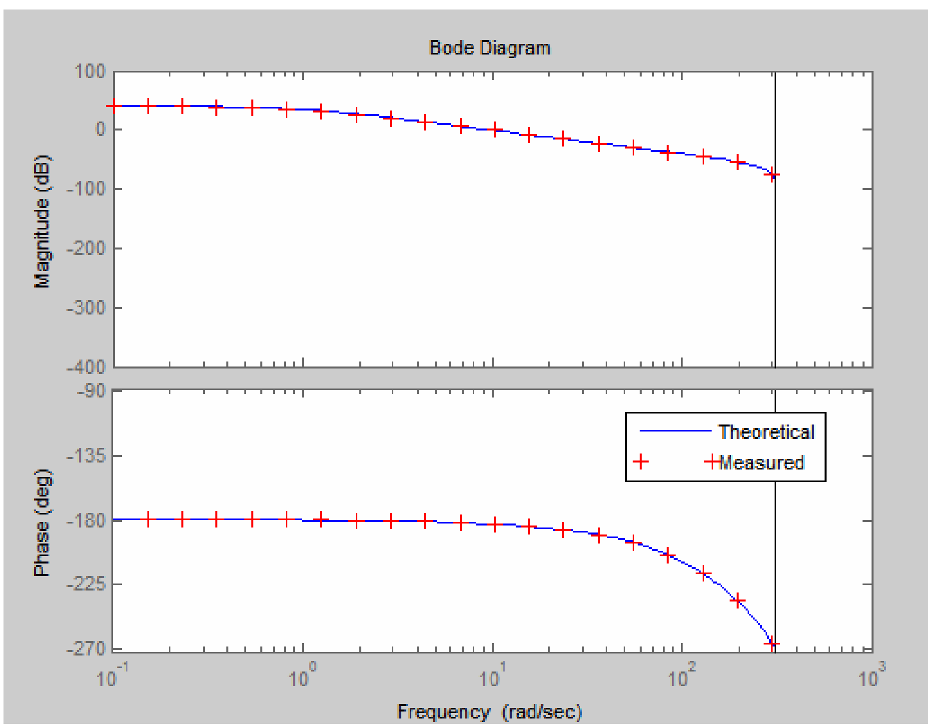

Figure 5: Plot of theoretical versus measured frequency response.

The Dynamic Signal Analyzer

The DSAGUI depends on the DSA block that was discussed in Watkins[1]. The DSA block, which is used to calculate G(jω), can be seen in Figure 6 and 7.

When the DSA function starts, the sine wave generator shown in Figure 6 outputs a sinusoid signal at a series of frequencies beginning at ω_i and finishing at ω_f. At each frequency, the DSA collects data from the two channels. Channel 1, as given in (3), is taken from the input of the plant. Channel 2, the output of the plant, can be represented by (4).

$$u(t) = A\sin(\omega t + \phi) \tag{3}$$

$$y(t) = A\left|G(j\omega)\right|\sin\left(\omega t + \phi + \angle G(j\omega)\right) \tag{4}$$

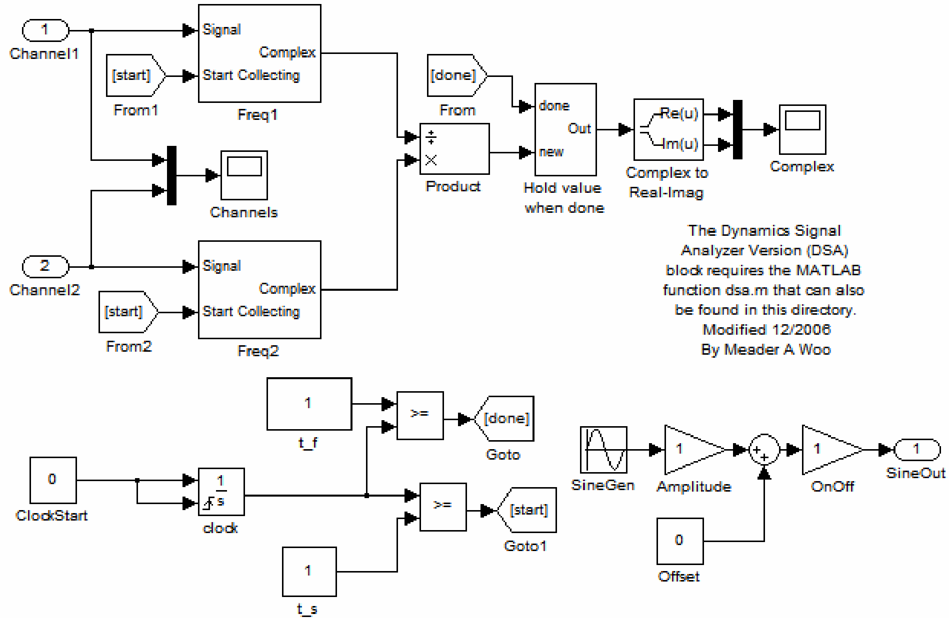

Figure 6: Inside the DSA Block

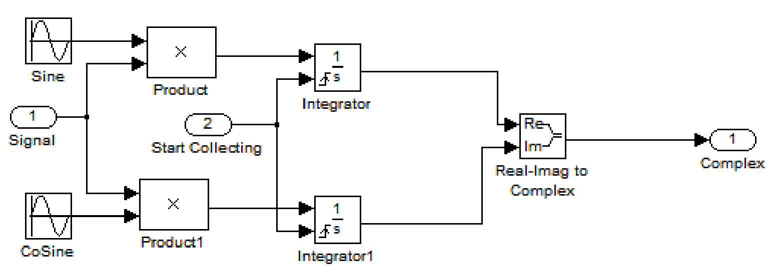

Figure 7: Inside the DSA/Freq1 block.

These two signals are fed into subsystems Freq1 and Freq2. Figure 7 shows subsystem Freq1, which is basically identical to Freq2. Then, by multiplying each individual equation/signal by $(\sin(\omega t) + j\cos(\omega t))$ and integrating over a number of periods we get equations (5) and (6).

$$X = A\cos(\phi) + jA\sin(\phi) = Ae^{j\phi} \tag{5}$$

$$Y = A\left|G(j\omega)\right|\cos\left(\phi + \angle G(j\omega)\right) + jA\left|G(j\omega)\right|\sin\left(\phi + \angle G(j\omega)\right) = A\left|G(j\omega)\right|e^{j(\phi+\angle G(j\omega))} \tag{6}$$

Dividing equation (6) by (5) we get $G(j\omega)$ which is given by (7) and shown in Figure 6.

$$\frac{Y}{X} = \frac{A\left|G(j\omega)\right| \cdot e^{j(\phi+\angle G(j\omega))}}{Ae^{j(\phi)}} = \left|G(j\omega)\right| \cdot e^{j(\angle G(j\omega))} \tag{7}$$

Using the DSA for Experimental Purposes

While the DSAGUI can provide the students in a classroom or a computer laboratory a significant amount of insight, it can provide them even more insight when combined with physical apparatus in the laboratory. A properly designed laboratory increases the students' "buy-in" to the concept, when they see it can actually be applied to a "real" system.

There are several different avenues for interfacing SIMULINK to the world outside the computer. Because of the ease of implementation, flexibility of approach, and the numerous experiments designed to work with the equipment, a platform from Quanser, Inc. was selected. The experimental environment incorporates physical plants, analog and digital sensors, A/D and D/A converters, and the WinCon real-time interface with SIMULINK. For more discussion of the laboratory setup, see the work by Watkins and O'Brien[4] and Esposito, Feemster, and Watkins[5].

In this example, we have a dc motor and we want use the DSAGUI to help us identify a continuous time model of the motor.

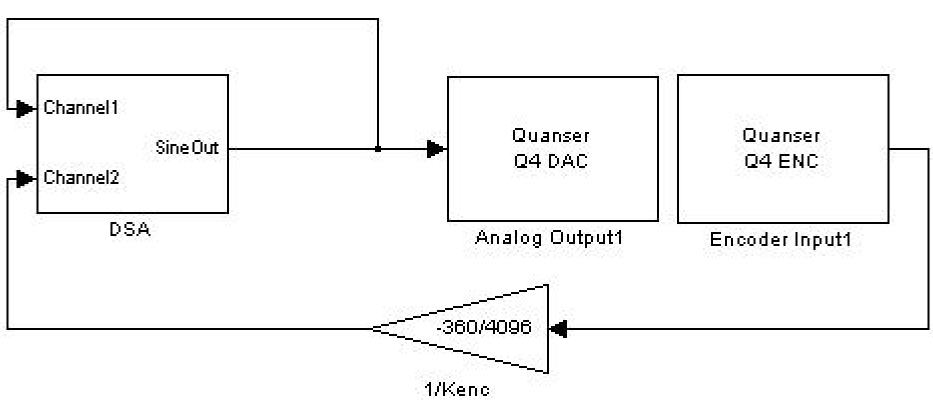

Figure 8: Using the DSA with Quanser Analog Input and Output Equipment.

Just as before, the DSAGUI's arguments are entered. Here we enter a frequency range of 1 rad/sec to 100 rad/sec with 20 frequency response measurement points, an estimated settling time of 0.2 seconds, and a sample period of 0.001 seconds. We also enter a signal amplitude of 4 and because this is a Wincon experiment the Wincon button will be activated as shown in Figure 9.

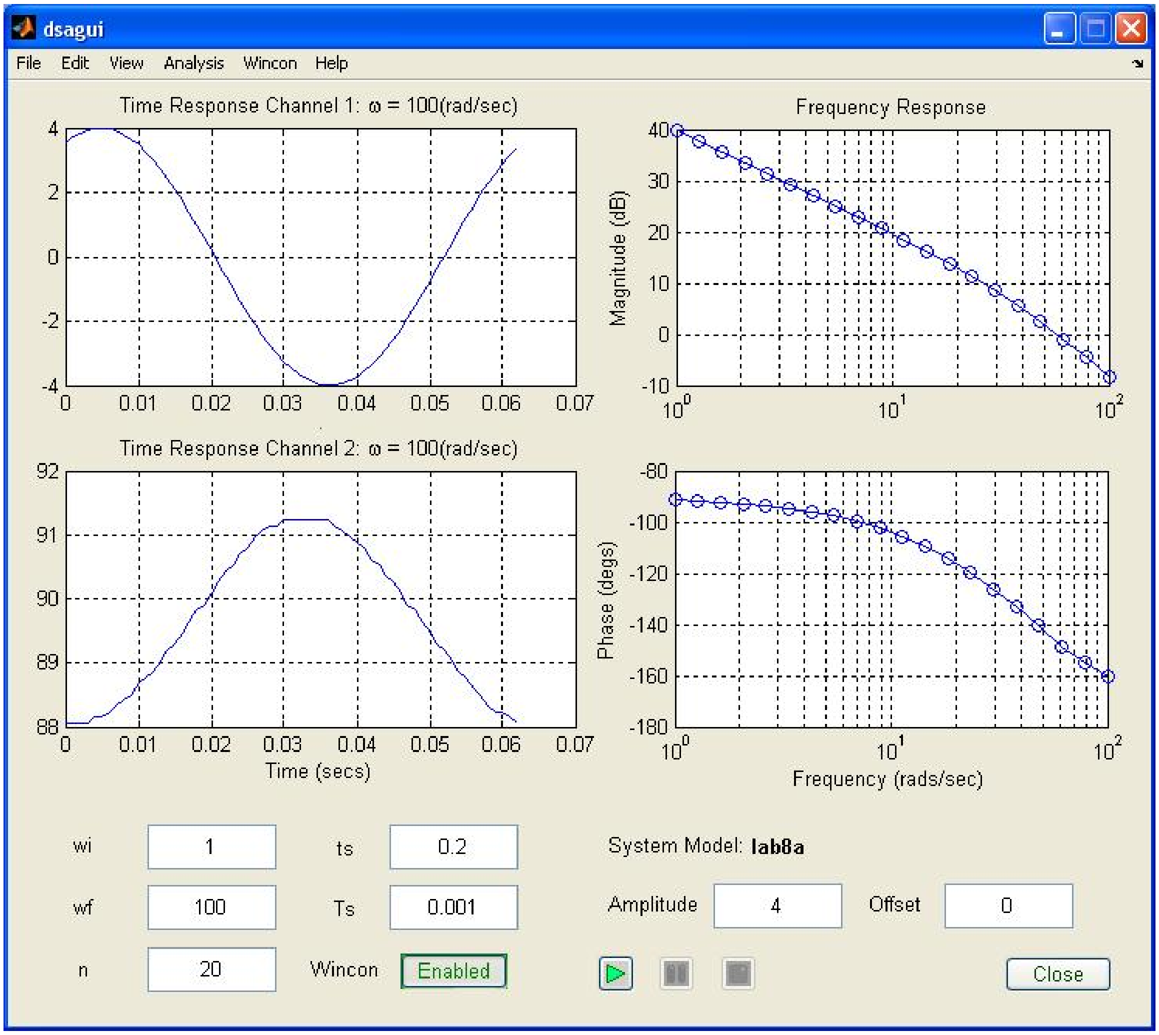

Figure 9: DSAGUI after a successful run using Quanser Wincon.

By exporting the frequency response, and taking the derivative of the results, the frequency response now looks like Figure 10. With a DC gain of 39.8dB and a 3 dB corner frequency of 40.8 rad/sec, the motor can be modeled using (8). A comparison of the measured and identified models is shown in Figure 11.

$$\frac{G(s)}{Ea(s)} = \frac{K}{s(s/\omega_c + 1)} = \frac{96.61}{s(s/40.8+1)}$$

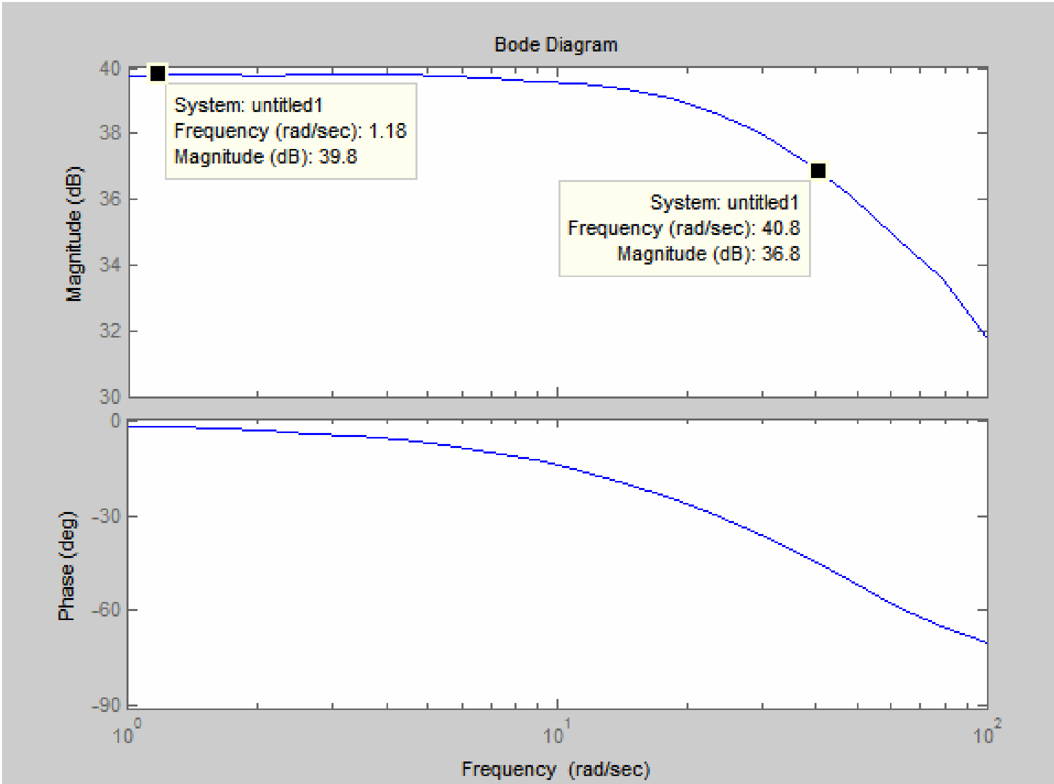

Figure 10: Analysis of the DSA results.

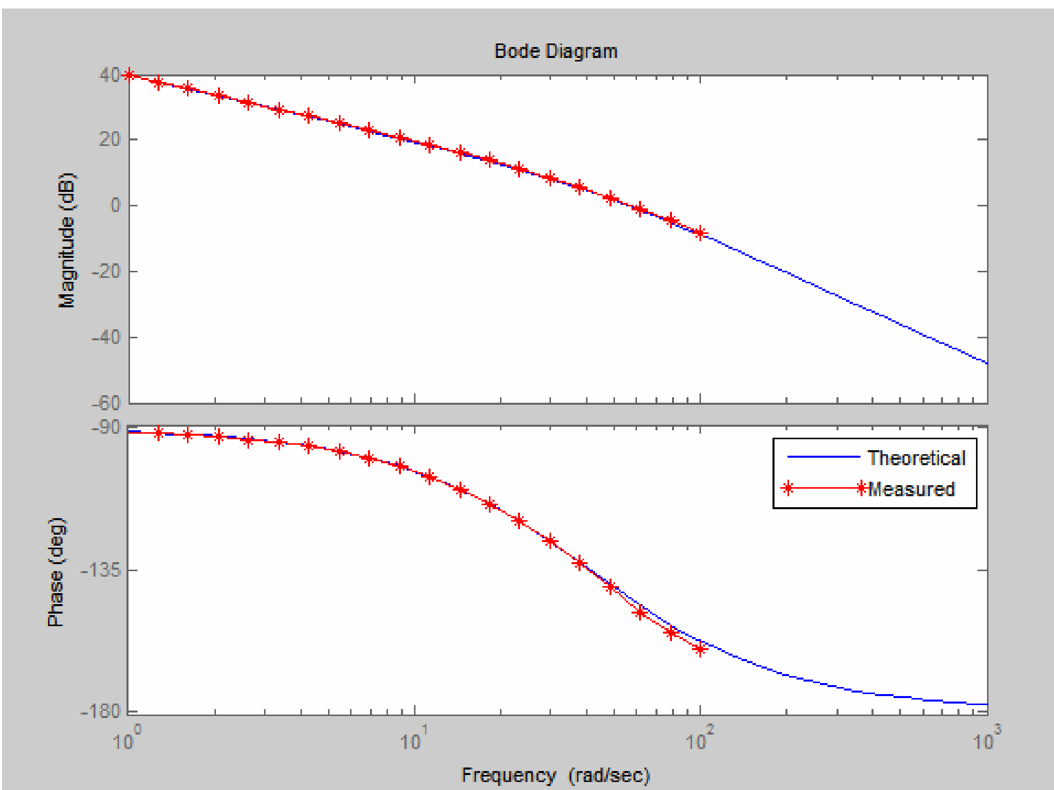

Figure 11: Plot of the measured frequency response and frequency response of identified (theoretical) model.

Conclusions

The ability to understand and utilize the frequency response of a linear system is a critical building block in many undergraduate engineering disciplines. Yet despite significant exposure to and need for the frequency response, many undergraduate students have limited intuition on what it means and represents. Furthermore, equipment for measuring the frequency response is often too expensive and complex for a typical undergraduate engineering lab. Consequently, we have developed a graphical user interface for a virtual dynamic signal analyzer that is used for measuring the frequency response. To run the DSAGUI in simulation mode requires only SIMULINK. However, if combined with the Quanser WinCon software and hardware input/output board, it can be used for measuring the frequency response of experimental apparatus in the laboratory.

In this paper, it has been shown how the DSAGUI can be used in the classroom and in the laboratory. A short discussion of how the DSAGUI works has also been included. The virtual DSA discussed in Watkins[1] has been used extensively in the control systems laboratory discussed in Watkins and O'Brien[4]. There are plans to replace the DSA with the DSAGUI during the upcoming academic year. The DSAGUI is available from the authors upon request.

References

[1] Watkins, J., "A Virtual Implementation Of A Dynamic Signal Analyzer Using Simulink," *Proceedings of the 2005 ASEE Annual Conference & Exposition: The Changing Landscape of Engineering and Technology Education in a Global World*, Portland, OR, June 2005.

[2] Wang, F., Abramovitch, D., and Franklin, G., "A Method for Verifying Measurements and Models of Linear and Nonlinear Systems," *Proceedings of the American Controls Conference*, San Francisco, CA, June 1993.

[3] Lilienkamp, K. and Trumper, D., "Dynamic Signal Analyzer for dSPACE," *Proceedings of the dSPACE User's Conference*, Dearborn, MI, May 2000.

[4] Watkins, J. and O'Brien, R., "A Novel Approach to a Control Systems Laboratory," *Proceedings of the 2003 ASME International Mechanical Engineering Congress & Exposition*, Washington, DC, November 2003.

[5] Eposito, J., Feemster, M. and Watkins, J., "Role of a MATLAB Real-Time Hardware Interface Within a Systems Modeling Course," *Proceedings of the 2004 American Society for Engineering Education Annual Conference & Exposition,* Salt Lake City, UT, June 2004.

MEADER WOO received his B.S. degree in electrical engineering from the Wichita State University 2004. He received his M.S. degree in electrical engineering from Wichita State University in 2006.

JOHN M. WATKINS is an Associate Professor in the Department of Electrical and Computer Engineering at Wichita State University. He received his B.S. degree in electrical engineering from the University of Nebraska-Lincoln in 1989 and his M.S. and Ph.D. degrees in electrical engineering from The Ohio State University in 1991 and 1995, respectively. His research interests include feedback control systems and time-delay systems.