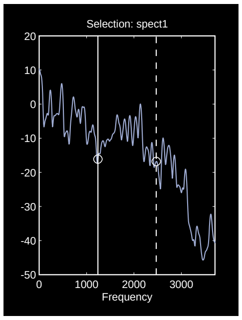
MATLAB® TECHNICAL COMPUTING BRIEF

Applications of Signal Processing



Abstract Two main pillars of signal processing technology are filter design and spectral analysis. This document describes and demonstrates through technical computing examples the application of MATLAB® to filter design and spectral estimation. These examples include the design of an IIR filter applied to a synthetic speech signal and the analysis of recorded speech data using the graphical interface tools included in the toolbox.

THE MATLAB® ENVIRONMENT

MATLAB® provides a powerful matrix environment for digital signal processing, system design, modeling, and algorithm development. Its accurate numeric computation and built-in visualization make working with signals, time-series data, and complex linear systems easy. The open system design of MATLAB and the application toolboxes enable you to change existing functions or add your own. MATLAB makes it possible to solve signal processing problems faster and more easily than with traditional programming languages such as C or Fortran.

Examples

SIGNAL PROCESSING TOOLBOX

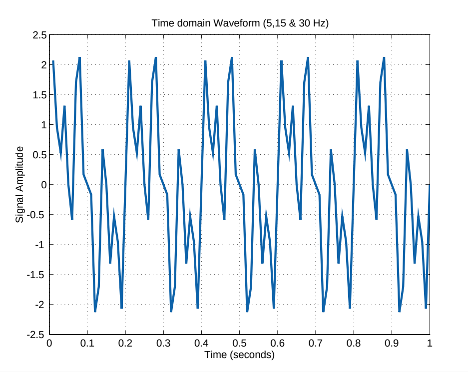
The Signal Processing Toolbox is a collection of MATLAB functions for analyzing, manipulating, and viewing signals and linear systems. The toolbox can be applied to many disciplines of engineering and science, including speech and audio processing, underwater acoustics, geophysics, communications, spectroscopy, and radar signal processing. Two main application areas addressed in the Signal Processing Toolbox are filter design and spectral analysis.

The following examples demonstrate the simplicity and power of using MATLAB and the Signal Processing Toolbox. In the first example, we use the command line functionality of the Signal Processing Toolbox to create a signal, design a digital filter, view the filter's frequency response, and use the filter to eliminate some of the frequencies from the generated signal. In the second example, we use the Graphical User Interface (`sptool`) of the Signal Processing Toolbox to pre-emphasize a signal by applying another filter and evaluating the resulting spectrum.

Filter Design Using the MATLAB Command Line

Signal Processing Toolbox supports a range of filter design methods, differing primarily in what parameters are specified. For example, one approach is to specify the order and cutoff frequency of the filter. A more rigorous approach is to specify the order, cutoff frequency, and ripple in the various bands. The filter might also be described in terms of frequency and magnitude vectors. All of these approaches are available in the Signal Processing Toolbox.

We know from Fourier analysis that signals can be described by a summation of frequency components. Typically, a filter is used to enhance signals by attenuating unwanted frequency components and retaining desired frequency components. In this MATLAB example, we begin by creating a signal `s` with three sinusoidal components (at 5, 15, and 30 Hz) and a time vector `t` of 100 samples with a sample rate of 100 Hz, and displaying it in the time domain. The MATLAB commands are shown in Figure 1. (Note: Text following the % sign denotes comments; remaining text is actual MATLAB code.)



```
% create 100 Hz time vector
Fs = 100;
t = (1:100)/Fs;
% create signal and plot
s = sin(2*pi*t*5)  + sin(2*pi*t*15) + ...
    sin(2*pi*t*30);
plot(t,s), grid
```

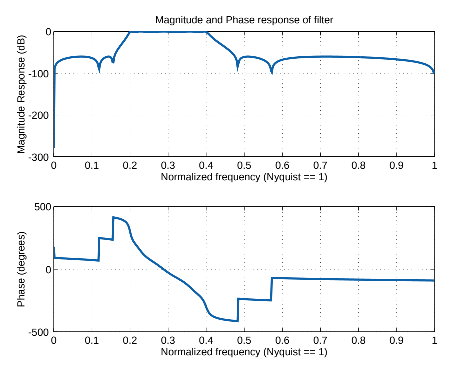
Figure 1. Signal with frequencies at 5, 15, and 30 Hz.

Now we design a filter to keep the 15 Hz sinusoid and eliminate the 5 and 30 Hz sinusoids. We use the functions `ellipord` and `ellip` to create an infinite impulse response (IIR) filter with a passband from 10 to 20 Hz. The `ellipord` function requires the specification of passband corner frequencies, minimum transition band frequencies near the passband corner frequencies, the maximum passband ripple in decibels (dB), and the minimum stopband attenuation in dB. In this example, we choose a transition frequency to be ±5 Hz near the passband corners, with a maximum of 0.1 dB ripple in the passband, and a minimum of 40 dB attenuation in the stopbands. We start by determining the minimum order (passband and stopband frequencies are normalized to the Nyquist frequency):

```
Wp1 = 10/50;  Wp2 = 20/50;  Ws1 = 5/50;  Ws2 = 25/50;
Wp = [Wp1 Wp2];  Ws = [Ws1 Ws2];  Rp = 0.1;  Rs = 40;
[n,Wn] = ellipord(Wp,Ws,Rp,Rs);
```

`ellipord` returns an order of 5, the minimum possible order for a lowpass prototype that will meet the constraints upon transformation to a bandpass filter. When we apply this order to the `ellip` function, internally we transform the lowpass prototype to a bandpass filter using the function `lp2bp`. This doubles the order, making `n = 10`.

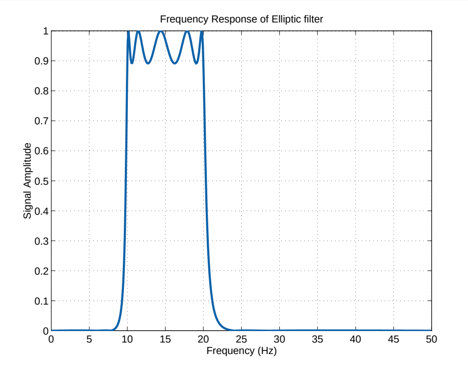
Next we use `n`, the order, and `Wn`, the passband corner frequencies, to actually design the filter. We also use `freqz`, a tool for computing and displaying the frequency response of the descriptive transfer function. When called with no left-hand-side arguments (i.e., return values), `freqz` displays the magnitude and phase response of the filter normalized to the Nyquist frequency, as shown in Figure 2a.



```
% determine filter coefficients
[b,a] = ellip(n,.1,40,Wn);
freqz(b,a)
```

Figure 2a. Magnitude and phase of filter, normalized to the Nyquist frequency.

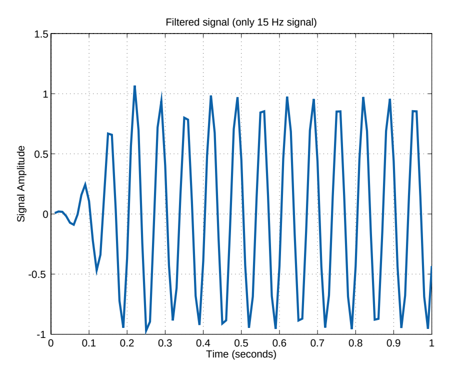
When called with left-hand-side arguments, `freqz` returns the filter's complex frequency response `H`, evaluated at 512 frequency points. Evenly spaced around the top half of the unit circle, the frequencies are returned in the vector `w`. Figure 2b shows how we plot the results after transforming the frequency vector.



```
% compute frequency response and plot
[H,w] = freqz(b,a);
plot(w*Fs/(2*pi),abs(H)),grid
```

Figure 2b. Magnitude response of the filter in Hz.

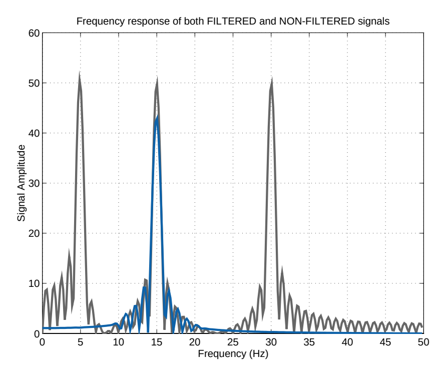
Like most filter design algorithms in this toolbox, the `ellip` function returns two vectors of coefficients that represent the filter as a linear system using the rational transfer function model. We now filter our original signal `s` and plot the time response in Figure 3 to see if we get the desired effect.



```
% filter and display time domain signal
sf = filter(b,a,s);
plot(t,sf)
grid
```

Figure 3. The final filtered signal. After initial transients, only a steady 15 Hz signal remains.

Except for a slight propagation delay through the filter, the filtered signal appears to be a single frequency sinusoid. Its frequency is determined by observing the power distribution generated with the following statements:



```
% original signal --> frequency domain
S = fft(s,512);
% filtered signal --> frequency domain
SF = fft(sf,512);
% scale the frequency vector
f = (0:255)/256*(Fs/2);
plot(f,abs([S(1:256)' SF(1:256)']))
grid
```

Figure 4. The frequency content of the signal before filtering (gray) and after filtering (blue). Note the suppression of the peaks at 5 and 30 Hz.

The final plot shown in Figure 4 displays the frequency content of the signal before and after the bandpass filtering. Notice that the filtered signal passes only 15 Hz, exactly as we had expected.

In this example, we will use both the Signal Processing Toolbox command-line functionality and the `sptool` Graphical User Interface to "pre-emphasize" a speech signal. The frequency range of human speech is from 0 to 4000 Hz. However, in the great majority of voices, there is very little speech present in the range above 2500 Hz. This artifact of speech is called *spectral tilt*. Many speech processing algorithms introduce noise that also occurs in this high-frequency range (above 2500 Hz). The addition of such high-frequency noise in conjunction with the reduction of speech in the same frequency range can mask the natural vocal sounds in that range, yielding poor results which are particularly evident after speech processing algorithms are applied. A technique called pre-emphasis is employed to enhance the frequencies in the high end of the spectrum. To explore this technique, we will examine the speech signal, design and apply a digital filter, and examine the results.

To start the example, we load a stored speech signal into the MATLAB workspace and then into `sptool` (see Figure 5). Note: this speech signal, `mtlb`, is included in all releases of MATLAB. It is a vector of size 4001 x 1 and has a sampling frequency of 7418 Hz.

```
load mtlb;
sptool;
```

After invoking `sptool`, we will load the `mtlb` signal into the `sptool` workspace by selecting **Import** from the **File Menu** to indicate that we are "importing" a signal from the MATLAB workspace and indicate the data signal and the sampling frequency (Fs = 7418 Hz).

The `mtlb` signal upon which we will operate, `sig1`, will be displayed in the menu. You can browse it by clicking on **View** to invoke the Signal Viewer (Figure 5).

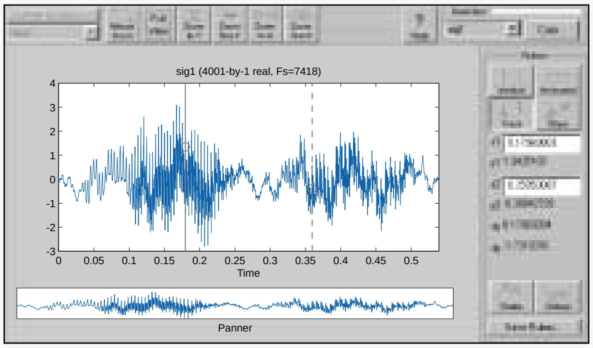


Figure 5. The Signal Viewer.

Once the `mtlb` signal is displayed in the Signal Viewer, you can examine it in many ways. Horizontal and vertical rulers are available for measuring the waveform. In addition, there are several types of "zoom" capability available for in-depth examination of the signal. Several signals or channels in different colors can be displayed at once for easy comparisons. You can also select **Play** from the **Options** menu to engage the sound card and hear the signal.

To evaluate the frequency content of the original signal, we use its spectrum. To create the spectrum, highlight the signal from which the spectrum will be created in the **SPTool** menu and select **Create** under the **Spectra** option. The Spectrum Viewer of the original `mtlb` signal is shown in Figure 6.

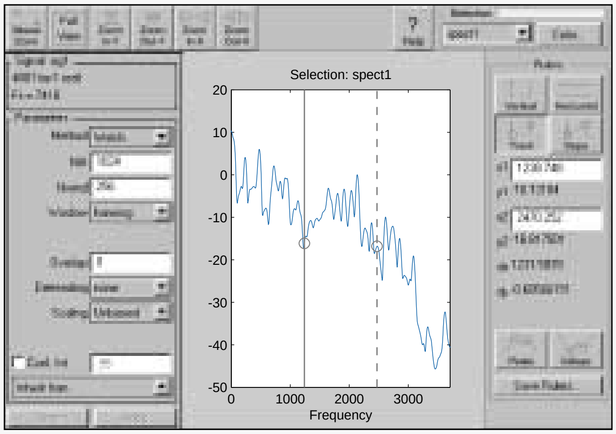


Figure 6. The Spectrum Viewer.

Four methods of spectral analysis are available in the spectrum viewer. Three of these are non-parametric: Welch's method, the MultiTaper Method, and the MUSIC (MUltiple SIgnal Classification) method. The fourth, the MEM (Maximum Entropy Method), is parametric. Parametric modeling techniques find the parameters for a mathematical model describing a signal, system, or process. The parametric methods use known information about the system to determine the model. Non-parametric methods assess the actual data itself and do not attempt to fit it to a model of any kind.

It is obvious from Figure 6 that the spectrum of the signal drops off sharply around the 2500 Hz frequency, thereby exhibiting spectral tilt. Like the Signal Browser, the Spectrum Viewer has rulers for both the *x* and *y* directions and zoom capability. All four of the spectral analysis methods discussed in the previous paragraph can be applied at any time to a spectrum within the Spectrum Viewer. Figure 6 shows Welch's method of spectral analysis applied to the signal.

We will design a high-pass filter to compensate for the spectral tilt using the Filter Designer GUI of `sptool`. The goal of this filter will be to emphasize those frequencies above the 2500 Hz range; therefore, our cutoff frequency will be 2500 Hz. An elliptic filter is chosen because in general, elliptic filters meet given performance specifications with the lowest order of any given filter type. In real-time applications, this will minimize memory usage and processor cycles.

The desired filter characteristics can be either typed in manually or indicated graphically. The filter parameters in the white box on the left side of Figure 7 show the correct parameters that need to be indicated for each type of filter. If you choose to enter them manually, you can type them directly into these fields. If you choose the graphical method, you can click and drag the mouse directly on the filter parameters displayed in gray in Figure 7. The filter order will be calculated and displayed with either method. Immediately after the filter parameters (cutoff frequencies f1 and f2, the passband ripple Rp, and the stopband ripple Rs) are entered, the filter graphic is updated to reflect them and our design is complete.

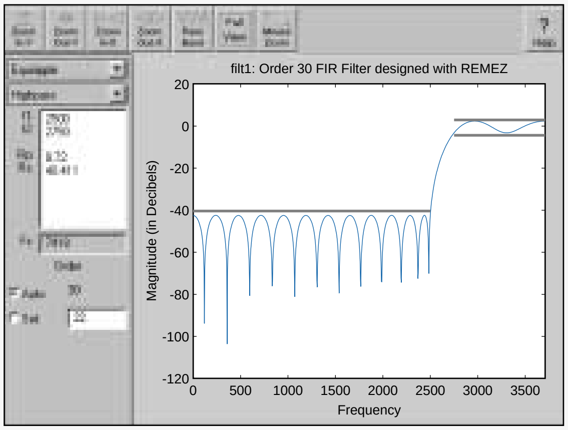


Figure 7. The Filter Designer.

Once the filter is designed, it can be applied to our input signal to create a new signal, `sig2`. To verify our experiment, a new spectrum will be created and compared to our original signal, `sig1`. This is accomplished in Figure 8. The spectrum of the original signal (`sig1`) is displayed in blue, and the spectrum of the filtered signal (`sig2`) is displayed in gray. The high-pass effect of our experiment has been successful—the spectrum no longer “tilts” down to the right; instead, it confirms that there is relatively greater energy in the frequency range that we chose to emphasize.

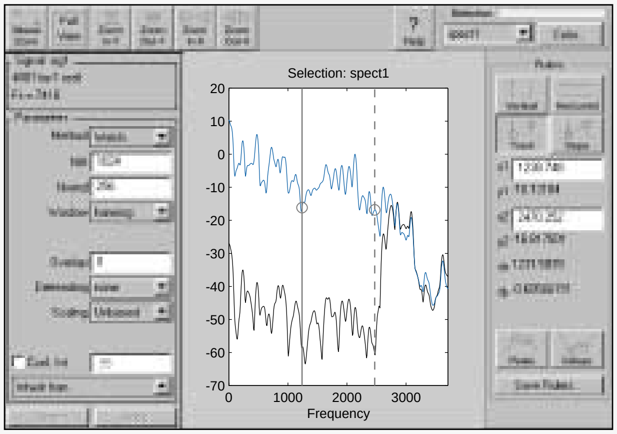


Figure 8. The Spectrum Viewer with original and filtered signals.

MATLAB Toolboxes

In addition to the Signal Processing Toolbox, The MathWorks offers more than a dozen application-specific toolboxes for use in signal processing applications. All MATLAB Toolboxes are implemented in the high-level MATLAB language, so you can modify the source code for functions or add your own. Simulink® and the DSP Blockset provide powerful block-diagram DSP design and system simulation tools for engineers developing DSP-based applications. You can easily combine the techniques in all of these products to design custom solutions for your specificproblems.

Toolboxes of interest to users of signal processing include:

Communications Toolbox
A collection of MATLAB functions and Simulink blocks for communications systems design, simulation, and analysis

Higher Order Spectral Analysis Toolbox
A collection of algorithms for analyzing non-Gaussian signals

Neural Network Toolbox
A collection of MATLAB functions for designing and simulating neural networks

System Identification Toolbox
An interactive environment for building accurate, simplified models of complex systems from noisy time-series data

Statistics Toolbox
Combines robust statistical algorithms with interactive graphical interfaces

Wavelet Toolbox
Powerful tools for signal and image analysis, compression, and de-noising

The MathWorks actively supports the ongoing education of students and professionals in signal processing and related fields. The MathWorks quarterly newsletter, *MATLAB News & Notes*, provides example applications and technical advice. Dozens of MATLAB based books have been written by world-renowned researchers and educators. These books use MATLAB to illustrate basic and advanced material in signal processing, control system design, math, physics, and other topics, and are used to teach courses in science, mathematics, and engineering. Many have accompanying diskettes with MATLAB scripts. The powerful combination of software and text helps both students and industrial users actively put theory into practice.

MATLAB Books

MATLAB based books for signal processing and system identification include:

Computer-Based Exercises for Signal Processing Using MATLAB
by C. Sidney Burrus, James H. McClellan, Alan V. Oppenheim, Thomas W. Parks,
Ronald W. Schafer, and Hans Schuessler
Prentice-Hall, 1994

Digital Filters and Signal Processing, 2e
by Leland Jackson
Kluwer Academic Publishers, 1989

Higher Order Spectral Analysis
by Chrysostomos Nikias and Athina P. Petropulu
Prentice-Hall, 1993

Principles of Signals and Systems
by Fred Taylor
McGraw-Hill, 1994

Discrete Random Signals and Statistical Signal Processing
by Charles W. Therrien
Prentice-Hall, 1992

System Identification: Theory for the User
by Lennart Ljung
Prentice-Hall, 1987

Identification of Linear Systems: A Practical Guideline to Accurate Modeling
by J. Schoukens and R. Pintelon
Pergamon Press, Oxford, 1991

For more information on MATLAB Toolboxes, as well as a list of MATLAB based books and ordering information, please contact your account representative at The MathWorks, Inc. at 508-647-7000.

For more information on Signal Processing, visit:

www.mathworks.com

Available MATLAB® Application Development Tools include:

- MATLAB language
- Debugger
- Editor
- Online Help Desk Reference
- M-file performance profiler
- Guide interactive GUI builder

The MathWorks DSP Workshop™ bundled package includes:

- MATLAB
- Signal Processing Toolbox
- Simulink®
- DSP Blockset

and optional components:

- Communications Toolbox
- Real-Time Workshop®

CONTACTING THE MATHWORKS

For inquiries outside the U.S. and Canada, contact your local distributor.

24 Prime Park Way
Natick, MA 01760-1500
U.S.A.

Tel: 508-647-7000
Fax: 508-647-7101

www.mathworks.com

- *info@mathworks.com*
  General Information
- *support@mathworks.com*
  Technical Support
- *service@mathworks.com*
  Customer Service
  order status, renewals, passcodes
- *suggest@mathworks.com*
  Product Enhancement Suggestions
- *resumes@mathworks.com*
  Resume Submission
- *access@mathworks.com*
  MATLAB Access Program
- *bugs@mathworks.com*
  Bug Reports
- *orders@mathworks.com*
  Place a Pre-paid Order
- *ftp.mathworks.com*
  FTP Server
- *comp.soft-sys.matlab*
  Usenet Newsgroup

International Distributors and Resellers

If your country is not listed, contact The MathWorks directly.

Australia
CEANET Pty., Ltd.
Tel: +61 (0)2-9922-6311
Fax: +61 (0)2-9922-5118
E-mail: *info@ceanet.com.au*
Web: *www.ceanet.com.au*

Brisbane office:
Tel: +61 (0)7-3369-4499
Fax: +61 (0)7-3369-4469
Authorized Distributors

Brazil
OpenCadd Computacao Grafica
Tel: +55-11-816-3144
Fax: +55-11-816-7864
E-mail: *opencadd@sanet.com.br*
Authorized Reseller

Czech Republic, Slovakia, Russia, Ukraine, Belarus, Moldavia
Humusoft s.r.o.
Tel: +420-2-68-44-174
Fax: +420-2-68-44-174
E-mail: *byron@humusoft.cz*
Web: *www.humusoft.cz*
Authorized Reseller

France
Scientific Software Group
Tel: +33-01-41-14-67-14
Fax: +33-01-41-14-67-15
E-mail: *info@ssg.fr*
Web: *www.ssg.fr*
Authorized Distributor

Germany, Austria
Scientific Computers GmbH
Tel: +49 (0)241-470-750
Fax: +49 (0)241-449-83
E-mail: *matlab.info@scientific.de*
Web: *www.scientific.de*

Unterföhring (Munich) office:
Tel: +49 (0)89-995-901-0
Fax: +49 (0)89-995-901-11
Authorized Distributors

India, Sri Lanka
Cranes Software International (P), Ltd.
Tel: +91 (0)80-5549-338
Fax: +91 (0)80-5546-299
E-mail: *matlab@CRANES.XEEBLR.xeemail.ems.vsnl.net.in*
Authorized Distributor

Israel
Omikron Delta (1927), Ltd.
Tel: +972 (0)3-561-5151
Fax: +972 (0)3-561-2962
E-mail: *info@omikron.co.il*
Web: *www.omikron.co.il*
Authorized Distributor

Italy
Teoresi s.r.l.
Tel: +39 (0)11-240-80-00
Fax: +39 (0)11-240-80-24
E-mail: *info@teoresi.it*
Web: *www.teoresi.it*
Authorized Distributor

Japan
Cybernet Systems Co., Ltd.
Tel: +81 (0)3-5978-5410
Fax: +81 (0)3-5978-5440
E-mail: *infomatlab@cybernet.co.jp*
Web: *www.cybernet.co.jp*
Authorized Distributor

Korea
Kimhua Technologies, Inc.
Tel: +82 (0)2-556-1257
Fax: +82 (0)2-556-4020
E-mail: *info@soft.kimhua.co.kr*
Web: *kimhua.co.kr*
Authorized Distributor

The Netherlands, Belgium, Luxembourg
Scientific Software Benelux B. V.
Tel: +31-(0)182-53-76-44
Fax: +31-(0)182-57-0380
E-mail: *info@ssb.nl*
Web: *www.ssb.nl*
Authorized Distributor

New Zealand
Hoare Research Software
Tel: +64-7-839-9102
Fax: +64-7-839-9103
E-mail: *info@hrs.co.nz*
Web: *www.hrs.co.nz*
Authorized Reseller

The Nordic Countries and Baltic States
Computer Solutions Europe AB
Tel: +46 (0)8-15-30-22
Fax: +46 (0)8-15-76-35
E-mail: *info@comsol.se*
Web: *www.comsol.se*

Søborg, Denmark office:
Tel: +45 (0)39-66 56 50
Fax: +45 (0)39-66 56 20
E-mail: *info@comsol.dk*
Web: *www.comsol.dk*

Helsinki, Finland office:
Tel: +358 (0)9-455-00-55
Fax: +358 (0)9-455-00-51
E-mail: *info@comsol.fi*
Web: *www.comsol.fi*

Trondheim, Norway office:
Tel: +47 (0)735-09-220
Fax: +47 (0)735-09-221
E-mail: *info@comsol.no*
Web: *www.comsol.no*
Authorized Distributors

Poland
ControlSoft
Tel: +48 (0)12-6-17-33-48
Fax: +48 (0)12-6-33-27-12
E-mail: *info@controlsoft.krakow.pl*
Web: *www.controlsoft.krakow.pl*
Authorized Distributor

Singapore, Malaysia, Thailand, The Philippines, Indonesia, Brunei
TechSource Systems Pte, Ltd.
Tel: +65-842-4222
Fax: +65-842-5122
E-mail: *info@techsource.com.sg*
Authorized Distributor

South Africa
Opti-Num Solutions
Tel: +27-11-325-6238
Fax: +27-11-325-6239
E-mail: *info@optinum.co.za*
Web: *www.optinum.co.za*
Authorized Reseller

Spain, Portugal
Addlink Software Cientifico
Tel: +34 (9)3 415-49-04
Fax: +34 (9)3 415-72-68
E-mail: *info@addlink.es*
Web: *www.addlink.es*
Authorized Distributor

Switzerland
Scientific Computers SC AG
Tel: +41 (0)31-954 20 20
Fax: +41 (0)31-954 20 22
E-mail: *info@scientific.ch*
Authorized Distributor

Taiwan
Scientific Formosa, Inc.
Tel: +886 (0)2-2505-0525
Fax: +886 (0)2-2502-4478
E-mail: *info@pent1.greenhills.com.tw*
Authorized Distributor

U.K., Ireland
Cambridge Control, Ltd.
Tel: +44 (0)1223-423-200
Fax: +44 (0)1223-423-289
E-mail: *info@camcontrol.co.uk*
Web: *www.camcontrol.co.uk*

Hove, England office:
Tel: +44 (0)1273-722-838
Fax: +44 (0)1273-720-550
E-mail: *pauline_fox@camcontrol.co.uk*
Authorized Distributors

8233v04 3/98