Composer
User Manual

for *SimplIQ* Servo Drives

Some changes have been made to the content in this manual and they will be incorporated into the upcoming release. To view the Addendum, click here.

March 2007

Elmo
Motion Control

Important Notice

This document is delivered subject to the following conditions and restrictions:

- This manual contains proprietary information belonging to Elmo Motion Control Ltd. Such information is supplied solely for the purpose of assisting users of the Elmo Composer software application, in conjunction with Elmo's *SimplIQ* line of digital servo drives.
- The text and graphics included in this manual are for the purpose of illustration and reference only. The specifications on which they are based are subject to change without notice.
- Information in this document is subject to change without notice. Corporate and individual names and data used in examples herein are fictitious unless otherwise noted.

Doc. No. MAN-COMPUM

Revision History

Ver. 1.9 **March 2007** Updated with Absolute Encoder Information (MAN-COMPUM.pdf)

Ver. 1.8 **September 2004** (MAN-COMPUM.pdf)

Ver. 1.6 **September 2003** (COMUGHA0903.pdf)
Support for analog incremental encoders (section 2.4), DC brush motors (section 2.7), new Auto Tuning dialog boxes (sections 2.8.2, 2.92 and 2.10.2) and enhanced motion monitor (section 3.2)

Ver. 1.5 May 2003

Ver. 1.0 January 2003

Elmo Motion Control Inc.
1 Park Drive, Suite 12
Westford, MA 01886
USA
Tel: +1 (978) 399-0034
Fax: +1 (978) 399-0035

Elmo Motion Control GmbH
Steinkirchring 1
D-78056, Villingen-Schwenningen
Germany
Tel: +49 (07720) 8577-60
Fax: +49 (07720) 8577-70

www.elmomc.com

Contents

Chapter 1: Introduction

The Composer is a sophisticated suite of Windows-based software designed by Elmo to enable you to quickly and easily set up and fine tune your motion control systems using Elmo's digital servo drives. The Composer can work with any brush or brushless servo motor.

You can use the Composer to:

- Tune the connected servo drive, either manually or automatically
- Test the controlled feedback system
- Interpret the test results and modify the test parameters
- Perform tests with different controllers to evaluate the closed-loop performance for different sytsem noise levels and different margins, in order to determine the optimal controller settings for specific applications

Each application created in the Composer includes a unique name, all host communication parameters used with the application, a dedicated program (if any), and all driver and motor parameters defined for that specific application. Each application can be applied to other drives with the same characteristics, without requiring a new setup process.

1.1 Composer Description

The Composer includes a range of tools for setting up your system. They include:

- The Composer **Wizard**, used for the initial tuning of single-axis servo drives. The Wizard tunes the drive to the motor, creates the application database and specifies the dedicated I/O components. It fully analyzes the entire system and defines all resonance and mechanical parameters.
- The **Smart Terminal**, used to manually manipulate the servo drive.
- The **Motion Monitor**, which controls the drive recording function to display the current drive status. The Motion Monitor can record and display almost any system value.
- The **Elmo Studio**, a basic environment for writing, downloading and executing programs in the connected drives.
- The **Application Editor**, which enables you to view all the parameters of the open application.
- The **Table Editor**, used to edit an existing PVT or PT table and to download it to the connected drive.
- The **Sync Manager**, which synchronizes the internal clocks of all drives connected through the CANopen network.
- The **Scope**, which displays the recorded parameters and provides a wide range of mathematical functions for manipulating the recorded curves. Data from the Scope can be exported to other programs such as Microsoft Excel or the MathWorks MATLAB.

- A **networking** option, which provides direct communication with multiple servo drives. You may set up as many as eight drives to communicate with the Composer via RS-232 communication, using different COM ports. In addition, you may connect up to 127 drives – each with a unique ID number – through CAN communication; the Composer supports the CANopen protocol for this type of network. After defining your network, you can save the configuration, and later restore the network in a single click.

In order to optimally configure your system, each data item must be correctly set and entered. The drive checks for database consistency before applying power to the motor, and rejects the data if inconsistencies are found. Once the motor is on, it prevents the acceptance of parameters that affect database integrity.

1.2 System Requirements

In order to install the Composer on your computer, the following items are required:

- Microsoft Windows 95/98/2000/XP
- At least 32 MB RAM
- 300 MB of hard drive space
- CD-ROM drive
- Pentium II processor or equivalent (minimum)
- RS-232 port
- CANopen board for CANopen serial communication (optional)

1.3 Composer Installation

You can install the Composer software either by using the CD-ROM delivered with the Elmo servo drive or by downloading the software directly from the Elmo website.

1.3.1 Installing from CD-ROM

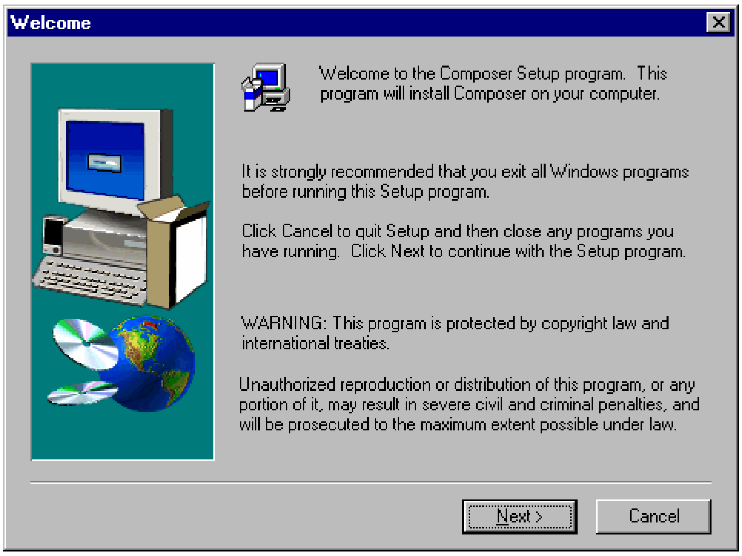
To install the Composer software from the CD-ROM to your hard drive:

1. Insert the CD-ROM into your CD-ROM drive.
2. From the Windows taskbar, select **Start - Run**. The Run dialog box will be displayed.
3. In the **Open** text box, type: **d:\setup.exe** or **e:\setup.exe**, according to the drive letter.
4. Click **OK**. The Install Wizard will be displayed.
5. Follow the Wizard instructions to install the Composer.
6. Upon completion, remove the CD-ROM from the drive.

1.3.2 Downloading from the Web

To install the Composer by downloading it from the Elmo website:

1. Using your Internet browser, go to the Elmo website: www.elmomc.com.
2. From the main menu, select **Support – Downloads**. The Support page will be displayed.
3. From the Support - Downloads menu, select **Software Tools** in the right column. The Software Tools page will be displayed.
4. From the Software Tools menu, select the **Composer, [date]** option in the **Description** column. The Windows File Download dialog box will be displayed.
5. Select the **Save this file to disk** option and click **OK**. The Save As dialog box will be displayed.
6. Navigate to the location where the Composer application should be stored and click **Save**. The software will be downloaded to that location.
7. After the download is complete (and you click **Close**), select **Start – Run** from the Windows taskbar. The Run dialog box will be displayed.
8. Click **Browse**, navigate to the Composer folder and select the Setup.exe file. Then click **OK**. The Composer Welcome dialog box will be displayed, as follows:



9. Click **Next** and follow the instructions to install the Composer.
10. Upon completion of the installation, you may wish to create a desktop shortcut for fast access to the application.

1.4 How to Use this Manual

This Composer user manual explains how to install the application, tune your Elmo servo drive and use the Composer software tools. It is organized as follows:

Chapter 2, *Using the Wizard*, explains how to connect your Elmo servo drive to your PC and the motor, and then tune the drive using the Composer Wizard.

Chapter 3, Using the Composer, describes the other Composer tools, including the Motion Monitor, the Smart Terminal, the Elmo Studio and the Application Editor.

Chapter 4, Using the Elmo Studio, describes the Elmo Studio program editing application integrated with the Elmo Composer.

The Appendix, *Using the Advanced Filter Designer*, explains how to use that tool to define a specialized filter used for tuning the velocity loop, position loop and dual loop.

The *Glossary* contains a list of terms used in the software, along with brief explanations.

This manual is an integral part of the *SimplIQ* documentation set, which includes:

- The Harmonica, Cello and Bassoon *Installation Guides,* which provides full instructions for installing one of Elmo's *SimplIQ* digital servo drives.
- The *SimplIQ Command Reference Manual,* which describes, in detail, each software command used to manipulate a *SimplIQ* digital servo drive.
- The *SimplIQ Software Manual,* which describes the comprehensive software used with a *SimplIQ* drive.

The following figure describes the accompanying documentation that you will require.

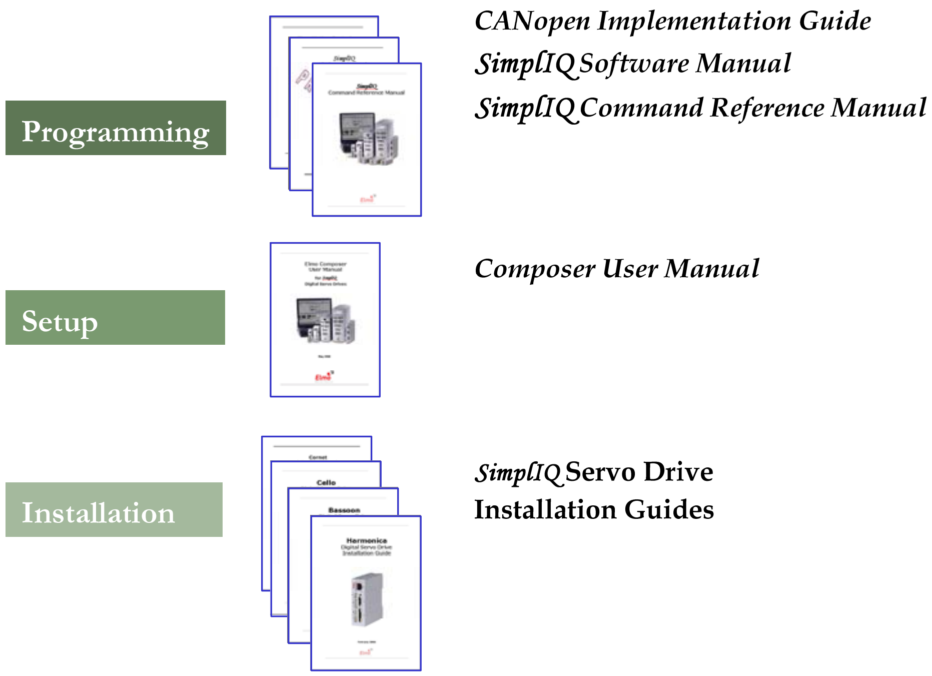


Figure 1-1: **Elmo Documentation Hierarchy**

Chapter 2: Using the Wizard

After connecting your Elmo servo drive to the motor and to the PC, you need to define its setup parameters in order to customize it to the motor, create the application (with the network, drive and motor parameters) and specify the dedicated I/O components. You can tune:

- A single-axis system
- A single dominant resonance mode, or two resonances that are far apart
- A balanced system, such as one with a *horizontal* axis

You may use the Composer Wizard to define the following:

- Motor parameters
- Commutation method and parameters
- Current loop tuning
- Commutation tuning
- Velocity loop tuning
- Position or dual loop tuning

Generally, the first time you use the Composer to initialize a drive, you will use the Wizard to define the drive application by tuning of the various loops either manually or automatically. The Composer stores the parameters for feedback, load, drive, communication and user program in an application database.

Once you have completed your initial setup using the Composer Wizard, you can verify that the configuration parameters meet your requirements by viewing them before running the motor. When you are satisfied with the configuration, you save the setup information as a designated application. This completes the installation and you can run the motor.

2.1 Before You Begin

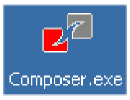
In order to ensure successful drive setup, you should verify that the following conditions are met:

- If you are using CANopen networking, be sure that the required CAN board(s) have been successfully installed.
- The static friction should be less than 20 percent of the full torque. While most systems use drives that can produce current satisfying this condition, it is recommended to check this by injecting 20% of full current and determining if the plant moves.
- The system should be properly balanced; that is, the motor speed should be 0 when zero current is injected to it.
- The system should be open-loop stable. For example, the Composer Wizard cannot be used with an inverted pendulum.
- The mechanical system should not have any low resonance below 5 Hz.

- The motor axes should be free to move plus or minus several electric poles.
- While the Wizard can operate with noisy or fairly inaccurate encoders, encoder accuracy should be no less than several hundreds per cycle.

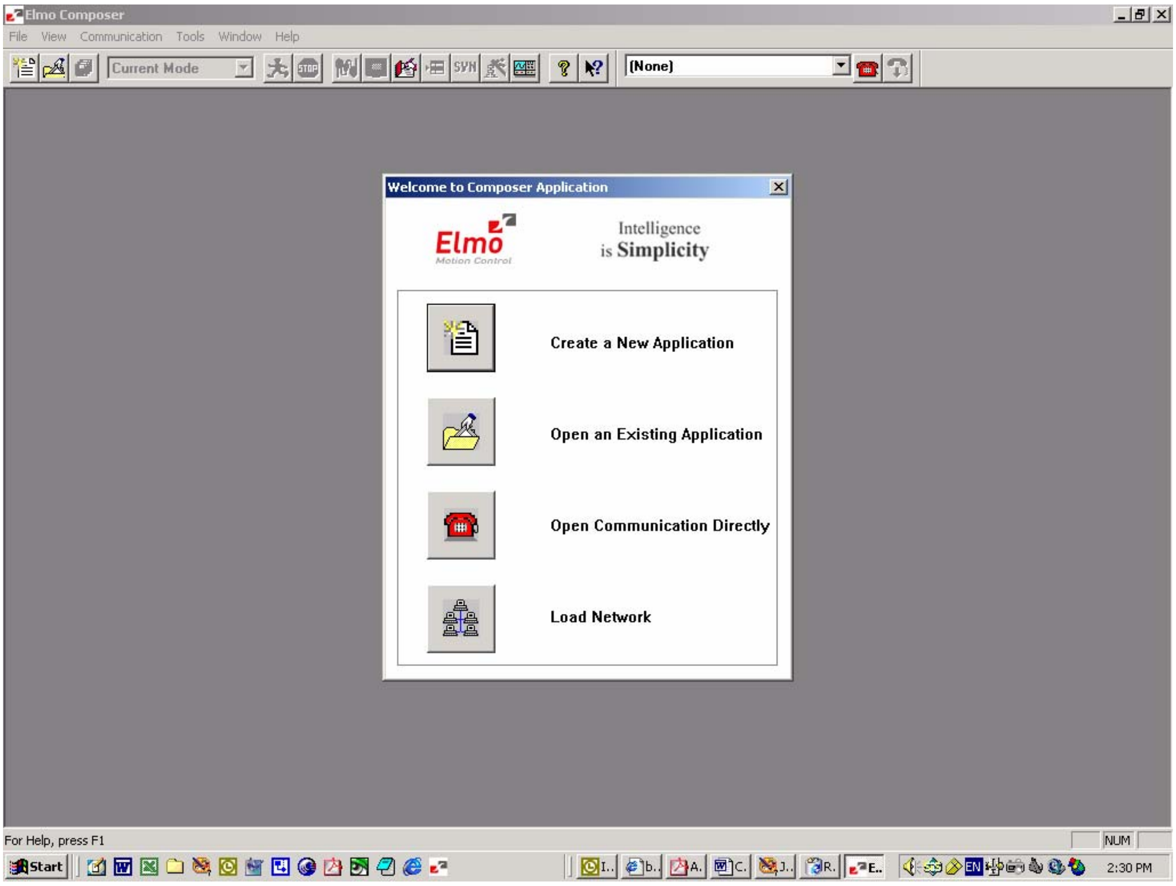
2.2 Accessing the Composer

You access the Composer in one of two ways:

- By clicking the 

 shortcut icon on the desktop
- By selecting **Start – Programs – Composer** from the Windows taskbar

The Welcome to Composer Application window is displayed immediately, as follows:

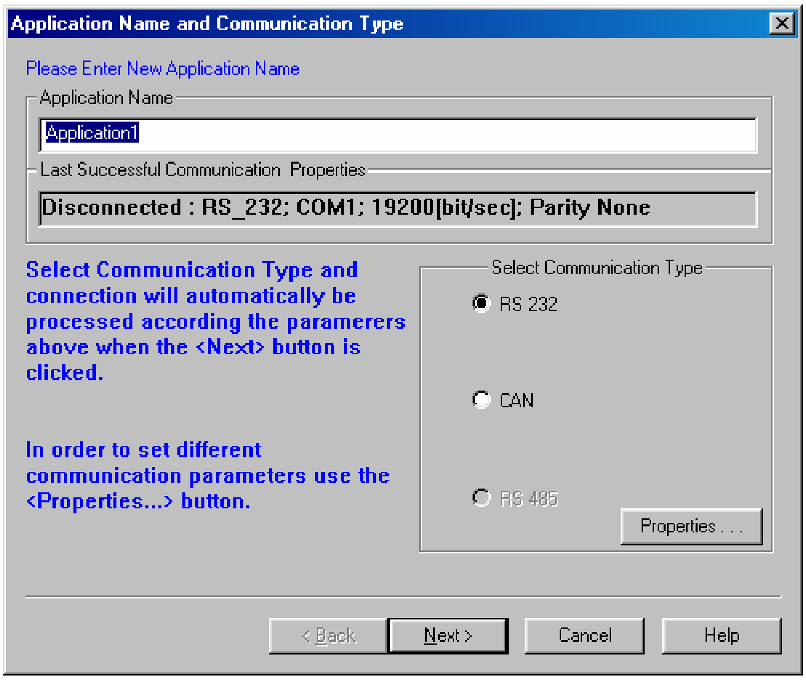


2.3 Creating a New Application

Creating a new application entails giving it a unique name, defining the communication network to be used and specifying the relevant motor and drive parameters. To define the new application, click **Create a New Application** from the Welcome to Composer Application window. If you have already accessed the Composer application, you can do one of the following:

- Click the  button in the Composer toolbar.
- Select **Tools – Create New Application** from the Composer menu bar.

The Application Name and Communication Type dialog box will be displayed, as follows:



This dialog box enables you to name your new application and to define the communication type used with it.

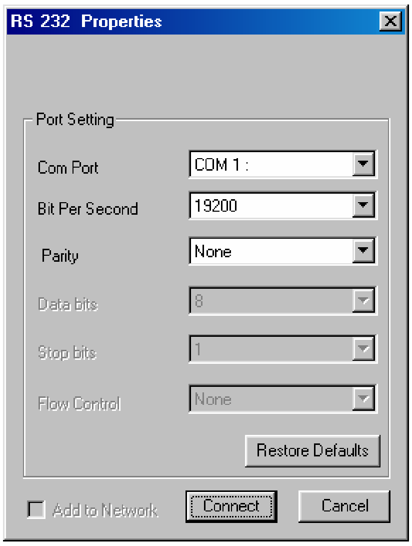
1. In the **Application Name** text box, type a name that clearly defines the new application. The parameters for the communications option last used by the host for the drive application are displayed in the **Last Successful Communication Propertie**s text box.

2. To display the parameters of a different communication type (RS-232 or CAN), click the relevant option in the **Select Communication Type** block. The parameters last used by the host for that type of communication (if any) will be displayed in the **Last Successful Communication Properties** text box.

3. If you are satisfied with the parameters displayed in the **Last Successful Communication Properties** text box, click **Next** to activate the connection and begin specifying your motor parameters (continue to section 2.4). If you wish to change any parameter for your selected communication type, click **Properties**. The relevant dialog box will be displayed (sections 2.3.1 and 2.3.2).

2.3.1 Defining RS-232 Communication

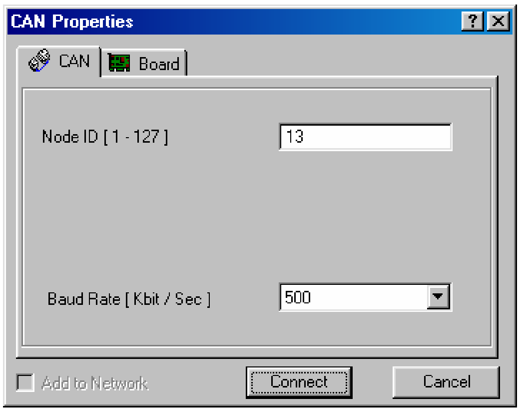
If you selected **RS-232** as your communication type, the RS-232 Properties dialog box will be displayed, as follows:



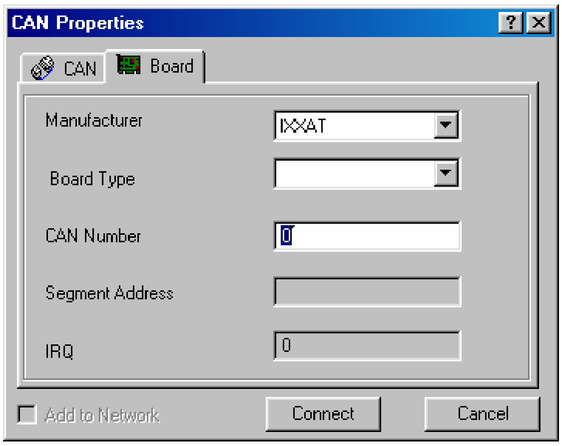
Use the **Com Port**, **Bits per Second** and **Parity** drop-down lists to select the appropriate setting for each parameter. You can use the **Restore Defaults** button to recall the default settings; the defaults are COM 1 (**Com Port**), 19,200 (**Bits Per Second**) and None (**Parity**). When you are finished, click **Connect** to activate the communication link. If a Communications Error occurs, try increasing the baud rate.

2.3.2 Defining CAN Communication

Before defining CAN communication parameters, be sure to define the specific CAN control board and firmware installed in your computer. Then select **CAN** as your communication type. The CAN Properties dialog box will be displayed, as follows:



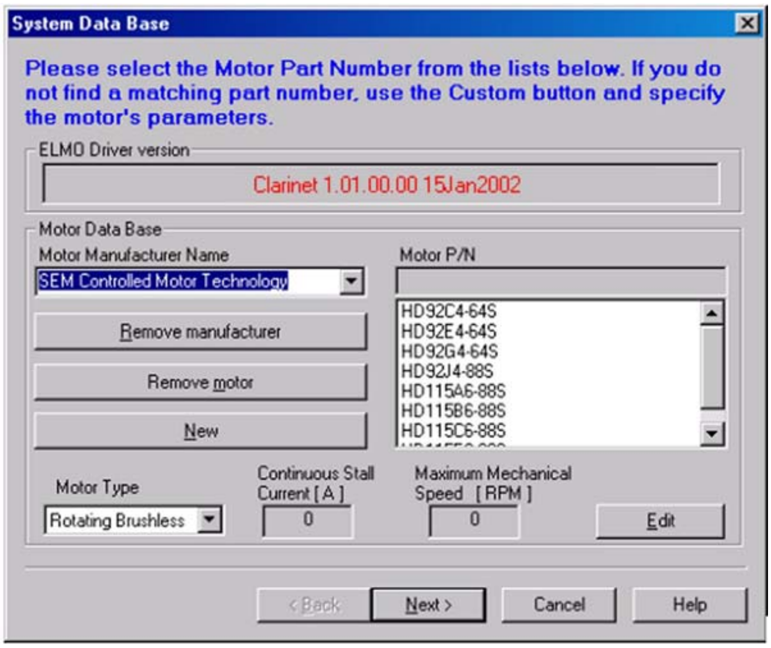
1. In the **Node ID** text box of the CAN dialog box, enter a unique number for the drive node. The permissible number of nodes ranges from 1 to 127.
2. From the **Baud Rate** drop-down list, select the baud rate that is used for all nodes of the network.
3. Click the **Board** tab to display the CAN Properties – Board dialog box, as follows:



4. From the **Manufacturer** drop-down list, select the name of the board manufacturer.
5. From the **Board Type** drop-down list, select the name of the CAN board installed in your computer.
6. In the **CAN Number** text box, type the number of the on-board controller (0 or 1) defined in the board setup process. The **Segment Address** and the **IRQ** value, defined during CAN installation, are displayed here only for reference.
7. Click **Connect** to activate the CAN network connection. This process may take a bit of time, and the Composer will display a status bar to indicate connection progress.
8.

2.4 Specifying the Motor Parameters

After defining your communication type and making the connection, the next step in drive setup is to define your motor parameters. You do this in the System Database dialog box, displayed when you click **Next** in the Application Name and Communication Type dialog box:



1. Check the Elmo servo drive name displayed in the **ELMO Driver version** text box (read from the controller) and verify that you are working with the correct drive.
2. This dialog lists motor manufacturers and motors that are stored in the drive database.

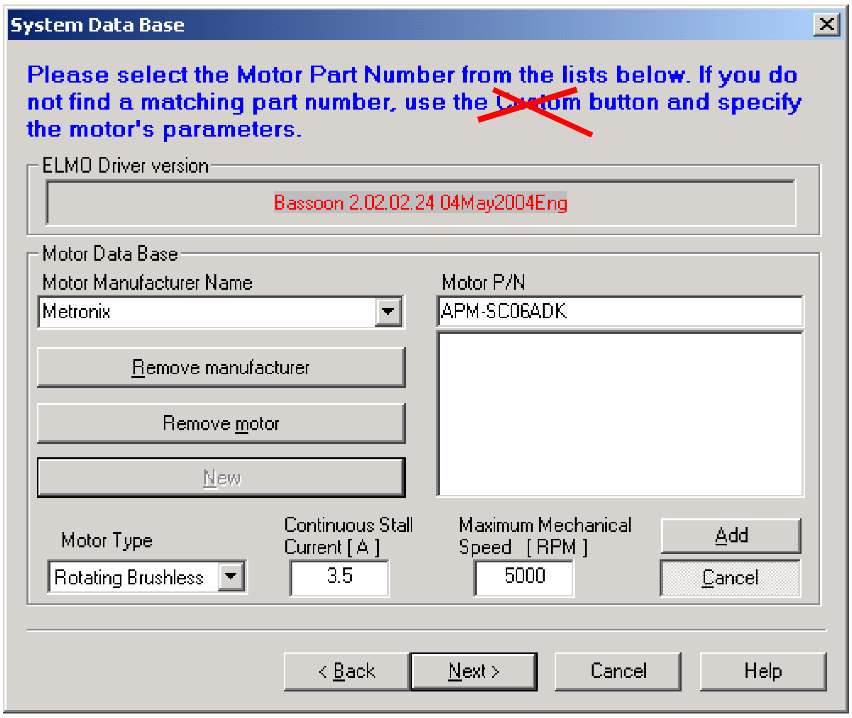
   If these lists *include* your motor, select the appropriate names from the **Motor Manufacturer Name** and **Motor P/N** lists. The motor parameters will be displayed in the lists and text boxes in the **Motor Parameters** block.

   If these lists do *not include* your motor, click the **New** command button, and in the **Motor Manufacturer Name** and **Motor P/N** text boxes**,** enter the appropriate information. Then, enter the following motor parameters:

   - **Motor Type**:
     For your specific type of Servo Drive, select **Linear Brushless**, **Rotating Brushless** or **Rotating Brush**.
   - **Continuous Stall Current**: the maximum allowed continuous motor current, in amperes (A).
   - **Maximum Mechanical Speed**: the maximum motor speed, in m/sec for linear motors and RPM for rotating motors.

   When entering the last two parameters, be sure that these two values are listed exactly as they appear in the motor manufacturer's data sheet.

   When you have finished creating the new entry for the database, click **Add**.

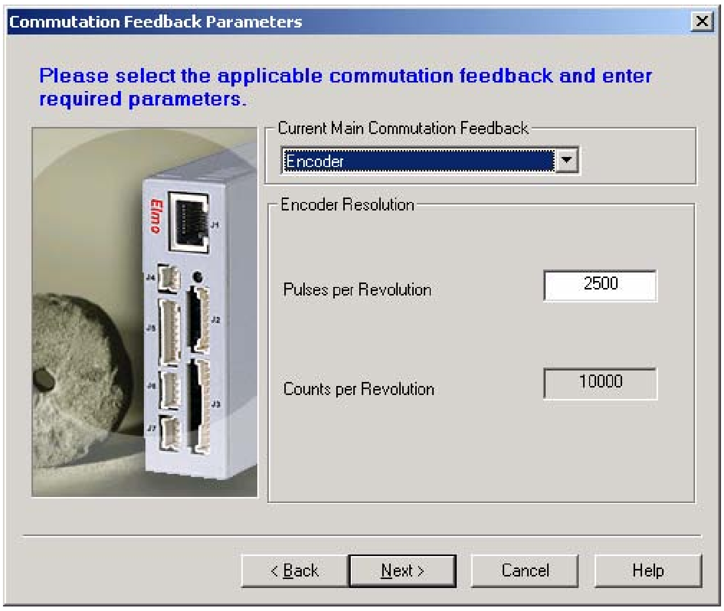


3. To change the motor parameters, select the motor and click **Edit**. You may use these changed values for testing in the Wizard. If you wish to save the new values, change the name in the **Motor P/N** text box and click **Add**.

   To *delete* a manufacturer name or a part number that you have entered, select the name and then click either **Remove manufacturer** or **Remove motor**, as appropriate. Only motors that you have entered in the database can be removed.

4. Click **Next** to continue defining your motor. The Commutation Feedback Parameters dialog box will be displayed.
5. From the **Current Main Commutation Feedback** drop-down list, select the primary encoder type used in your motor (the system may also have an auxiliary feedback mechanism). According to the motor type defined in the System Database dialog box, the system will display the text boxes relevant to defining the current commutation feedback.

For a rotary motor, a dialog box similar to the following will be displayed:



For a linear motor, a dialog box similar to the following will be displayed:

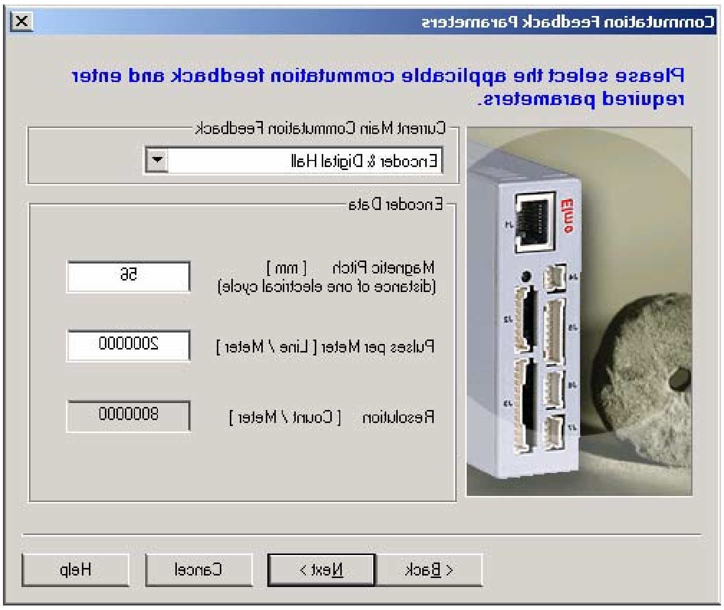


Table 2-1 outlines the various options (parameters are fully explained in the Glossary of this manual).

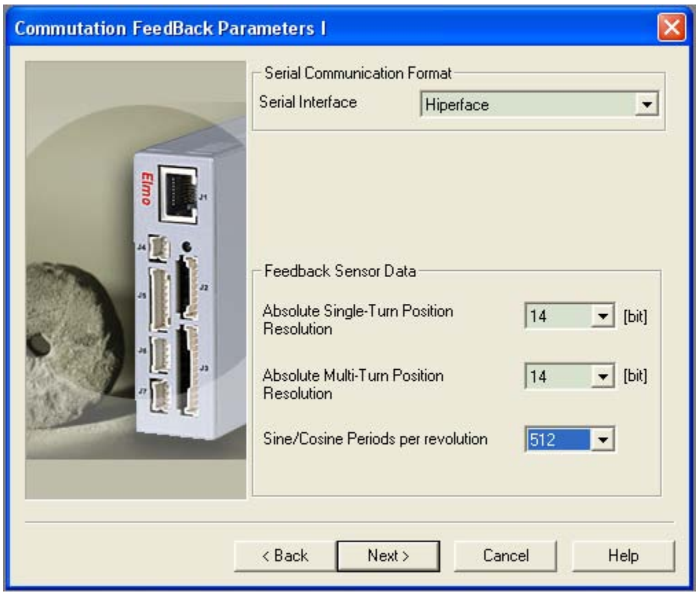
Motor Type	Current Main Commutation Feedback	Text Box
Linear Brushless	Encoder *or* Encoder and Digital Hall	**Magnetic Pitch** (m) **Pulses per Meter** (lines/m) **Resolution** (counts/m) (4 x Pulses per Meter value)
Rotating Brush	Encoder	**Pulses per Revolution** **Counts per Revolution** (4 x Pulses per Revolution value)
Rotating Brushless	Encoder *or* Encoder and Digital Hall	**Pulses per Revolution** **Counts per Revolution** (4 x Pulses per Revolution value)
	Digital Hall	**Number of Pairs of Poles**
Linear DC	Encoder	**Pulses per Revolution**

Table 2-1: ***Current Main Commutation Feedback Parameters***

6. Enter the values for each of the displayed parameters.
7. Click **Next** to complete the motor definition and continue to the next stage of setup.

2.5 User Interface for Absolute Feedback

2.5.1 Rotation Motors- Heidenhain and Stegmann



- **Serial Interface–** a drop down menu with the following options: Hiperface or EnDat2.1 formats.

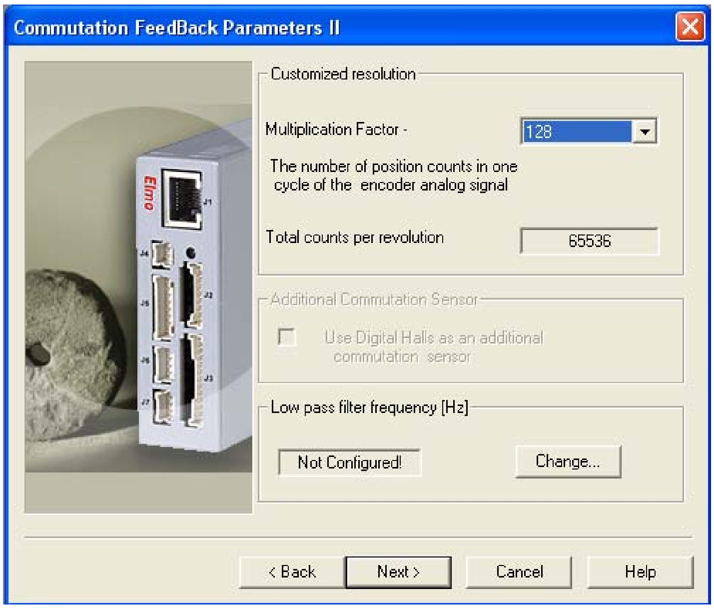
- **Absolute Single Turn Position Resolution**–a drop down menu that represents, in terms of bits, single-turn feedback cycle. The number of counts would be 2^number of bits. This value is practically the digital absolute resolution derived from the feedback and can be read from the encoder data sheet. Sine/cosine resolution improves this value but is derived from firmware manipulation. The value can be read from the encoder data sheet.
- **Absolute Multi-Turn Position Resolution**– a drop down menu that represents, in term of bits, the multi-turn feedback resolution. The number of full mechanical resolution without loosing the origin (absolute position) is 2^number of bits turns.
- **Sine/Cosine Periods per Revolution**– a drop down menu that represents the number of sine/cosine signal cycles per mechanical revolution. After the absolute position is determined by the *SimplIQ* drive (typically after power up), the position is calculated from the sine/cosine signals and the "Multiplication Factor".

  Note:

  The following should apply:

  1. *SINGLE-TURN* + *MULTI-TURN* ≤ 32 bit resolution
  2. *SINGLE-TURN* ≤ 32 and *MULTI-TURN* ≤ 16 bit resolutions

  By clicking on the "Next" button, the "Commutation Feedback Parameter II" window will appear. The Composer updates the drive with the relevant commutation and feedback parameters needed.



- **Multiplication Factor** – is a drop down list that represents, in term of bits, the number of position counts in one cycle of the analog (Sine\Cosine) signal. The limitation for the multiplication factor are:

  For rotary motors:

$$Multiplication\ \ Factor \leq \frac{10^9}{(2^{MULTI\_TURN}+1)*N}$$

  Where:

N: Number of sine/cosine signals per mechanical resolution

For linear motors:

$$Multiplication\ \ Factor\ \leq \frac{10^9 * N}{L}$$

where:

N: Period of a single analog sine\cosine. Typically in mm/inches

L: Length of the linear sensor. Typically in meters/legs

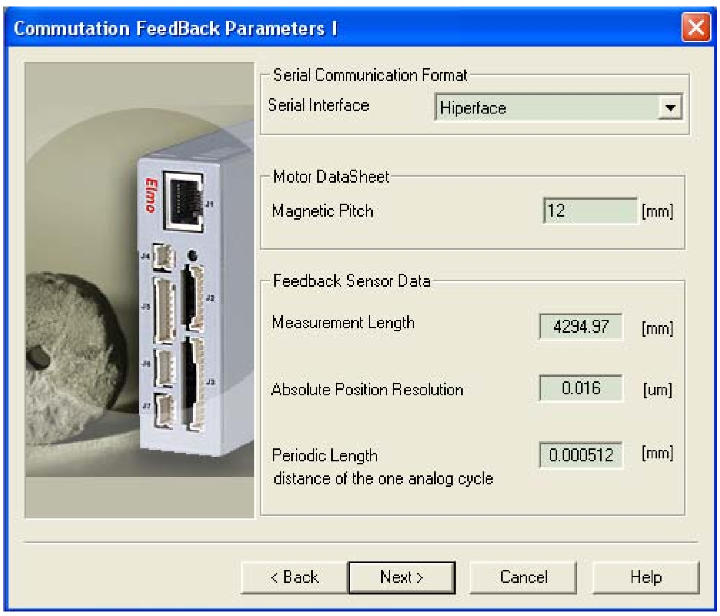
Both N and L must be the same (distance) units.

- **Total counts per revolution**– displays the final calculated user resolution
- **Additional Commutation Sensor-** The commutation sensor in absolute encoders is derived from the analog signals (sine/cosine). In cases where the sensor is not located on the motor and the motor is mounted with Digital Halls, the "Additional Commutation Sensor" enables correction of the commutation angle in each Hall edge. The feature is available for the Cornet, Tuba, Cello, Didge, Eagle, Falcon and Drum products for rotary and linear motors.
- **Change…-** The push button introduces the "Low Pass Filter" window. The low pass filter filters the analog signals for Speed-Readout. By default there is "no value" and "Not Configured!" appears. By clicking on "Change…", the user can apply any filter and see the theoretical step response of his/her choice. Typical values for this filter are between 400-700Kz.

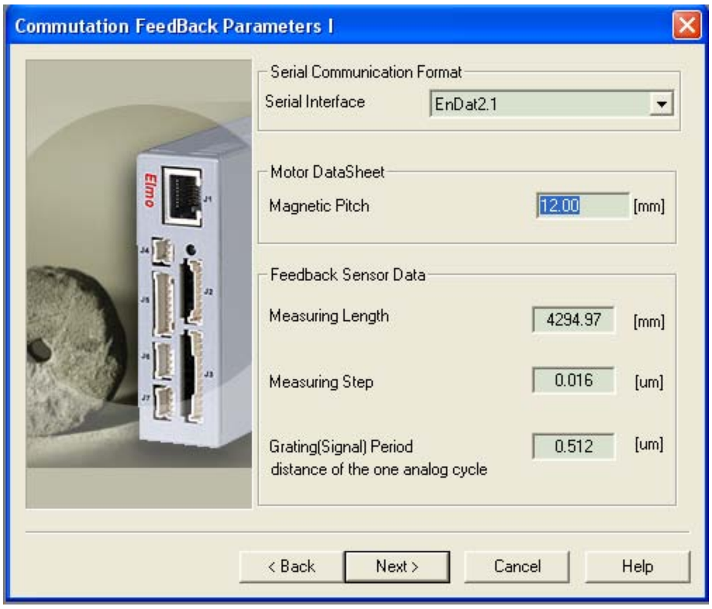
Click on the "Next" …

2.5.2 Linear Motors

For Stegmann feedbacks:



For Heidenhain feedback:



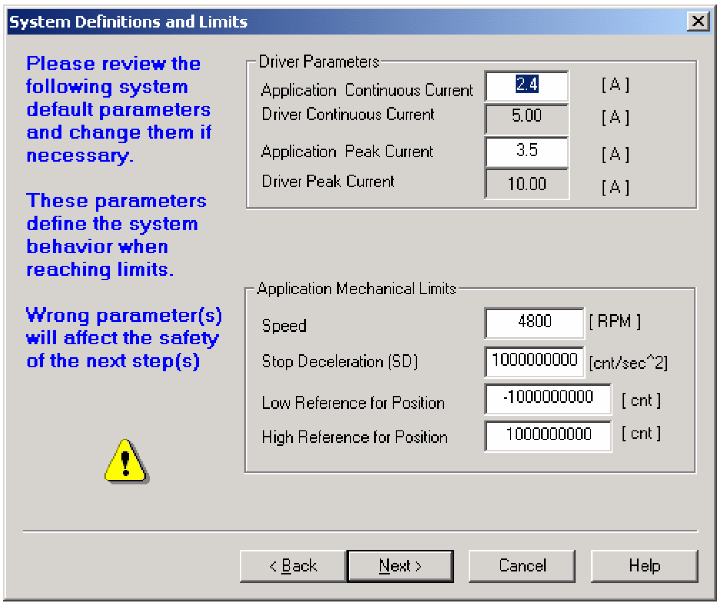
- **Serial Interface–** a drop down menu with a list of the serial interfaces: Hiperface or EnDat2.1 formats.
- **Magnetic Pitch**– an Edit Box which represents the distance of one electrical cycle in millimeters. The value is taken from the motor's data sheet.
- **Measurement Length (Hiperface) or Measuring Length (EnDat)**– an Edit Box which represents the length of linear sensor in millimeters. The value is taken from the sensor's data sheet.
- **Absolute Position Resolution (Hiperface) or Measuring step (EnDat)**– an Edit Box which represents the distance between two sequential absolute position readouts. The value is taken or calculated from the sensor's data sheet.
- **Periodic length (Hiperface) or Grating (Signal) Period (EnDat)**- an Edit Box which represents the distance of one analog signal cycle. The value is taken from the sensor's data sheet.

When clicking on the "Next" button, the "Commutation Feeeback Parameter II" window appears.

- **Position Counts per Meter**- an Edit Box which represents the number of position counts per meter. Integer type.
- **Change… -** This push button introduces the "Low Pass Filter" window. The low pass filter filters the analog signals for Speed-Readout. By default there is "no value" and "Not Configured!" appears. By clicking on "Change…" the user can apply any filter and view the chosen theoretical step response. Typical values for this filter are between 400-700Kz.

2.6 Defining System Limits

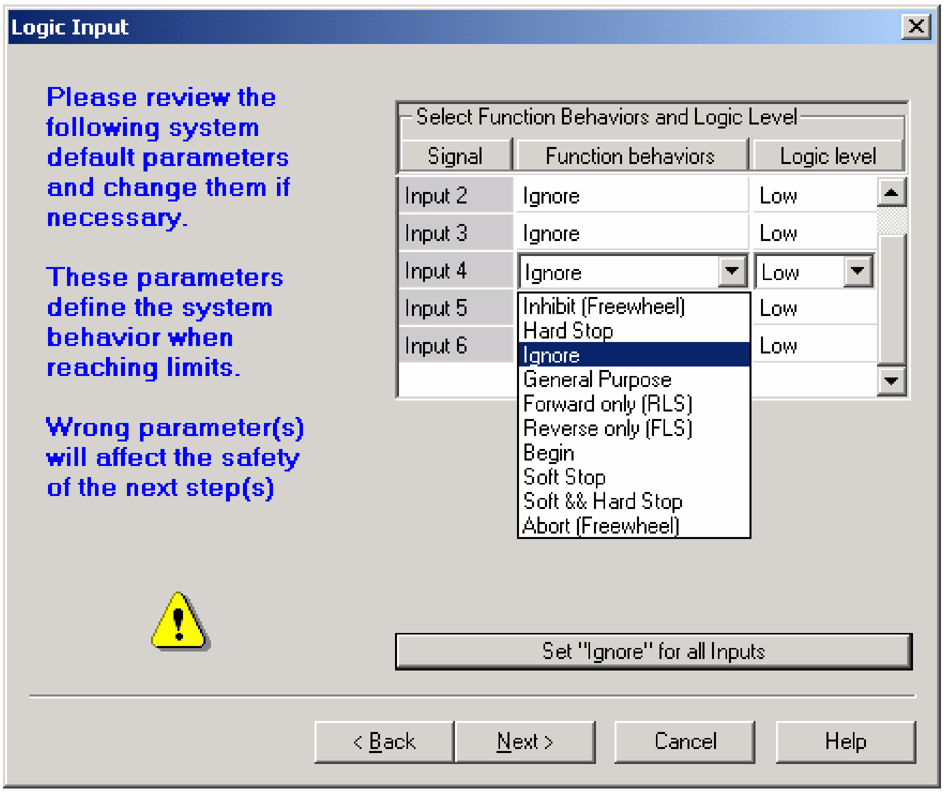
After defining your motor and commutation parameters (clicking **Next** in the Commutation Feedback Parameters dialog box), the System Definitions and Limits dialog box will be displayed to enable you to define how your system should behave when it reaches an operational limit. You need to specify each value; these parameters are **NOT** defined automatically by the system. Be sure that the parameters are correctly defined because incorrectly set values could affect system safety.



1. In the **Driver Parameters** block, enter the following values:
   - **Application Continuous Current**: the maximum current, in amperes, to be used with the connected motor. This value must be equal to or less than the **Motor Continuous Stall Current** (defined in the System Database dialog box, section 2.4) *and* the value displayed in the **Driver Continuous Current** text box.
   - **Application Peak Current**: the maximum short-term current, in amperes, that can be used with the application. The Wizard automatically displays the **Driver Peak Current** defined by the manufacturer. The application peak current value must be equal to or less than this value.
2. In the **Application Mechanical Limits** block, enter the following values:
   - **Speed**: the maximum motor speed that will be used in the application. This parameter affects the reference speed limits and must be equal to or lower than the **Maximum Mechanical Speed** value defined for the motor in the System Database dialog box (section 2.4).
   - **Stop Deceleration (SD)**: the maximum rate at which the load may decelerate.
   - **Low Reference for Position** and **High Reference for Position**: the full range, in counts, to be used for determining the absolute position command.

> **Note:**
> Entering a number higher than 1,073,741,822 indicates that there are no mechanical limits.

3. Click **Next**. The Logic Input dialog box will be displayed.



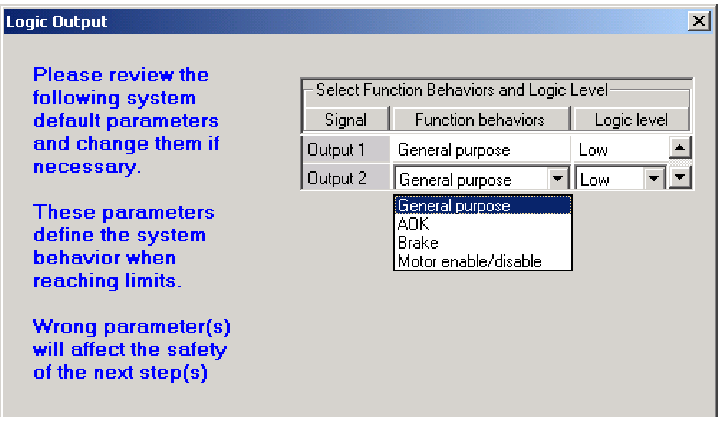
This dialog box enables you to define the actions that should occur when the various input signals – are activated.

4. From the **Select Function Behaviors and Logic Level** list, select the relevant function options and the logic level at which they are activated. The following table explains the options available for each of the switches:

Function Behavior	Description when Activated
Inhibit (Freewheel)	The drive shuts down and the motor runs freely.
Hard Stop	The motor stops under hardware control.
Ignore	Uncommitted input. No special function is launched.
General Purpose	The relevant general purpose function is launched.
Forward Only (RLS)	The motor moves only in forward motion.
Reverse Only (FLS)	The motor moves only in reverse motion.
Begin	The motor begins operating.
Soft Stop	The motor stops under software control.
Soft & Hard Stop	The motor stops under both hardware and software control.
Abort (Freewheel)	The drive shuts down and the motor stops.
Home	Main home option is launched. *Available only in Input 5 list.*
AUX Home	Auxiliary home option is launched. *Available only in Input 6 list.*

Table 2-2: ***Function Behavior Options - Input***

5. Click **Next** to continue to the Logic Output dialog box.

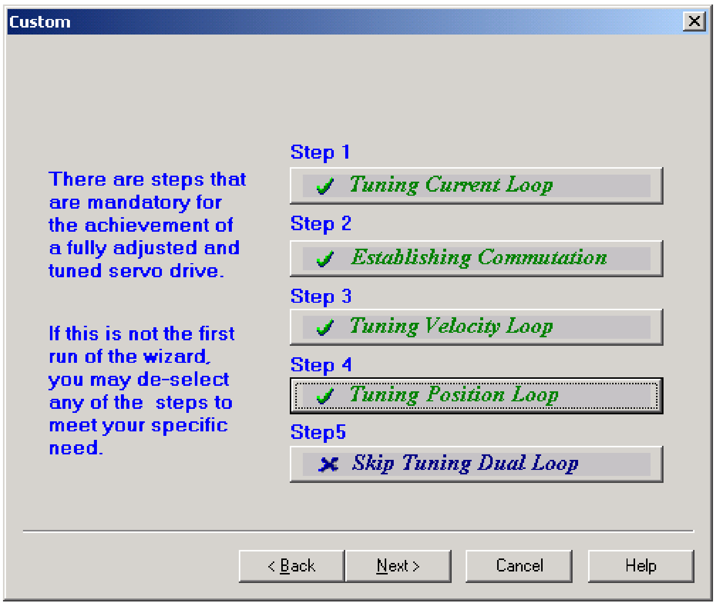


6. In the Logic Output dialog box, repeat the selection process for the output signals, listed in the following table:

Function Behavior	Description when Activated
AOK	The drive is ready for use.
Brake	The brake is engaged.
General purpose	Uncommitted output.
Motor enable/disable	Enable or Disable the motor.

Table 2-3: ***Function Behavior Options – Output***

7. Click **Next** to continue to the tuning steps. The Custom dialog box will be displayed.



If this is a first-time configuration, the Composer Wizard automatically selects the steps that are required for fully tuning the drive and motor that you have defined. If you selected a **Rotating Brush** motor type in the System Database dialog box (section 2.4), steps 1 and 2 will automatically be deselected. Steps 4 and 5 are mutually exclusive; you may select one or the other, or choose to skip them entirely.

To skip a step, click on the command button to toggle it to **Skip . . .** (the option turns blue).

If you have not performed commutation tuning yet or if you have changed any parameters in the Commutation Feedback Parameters dialog box, you *must* perform this step; you may not skip it.

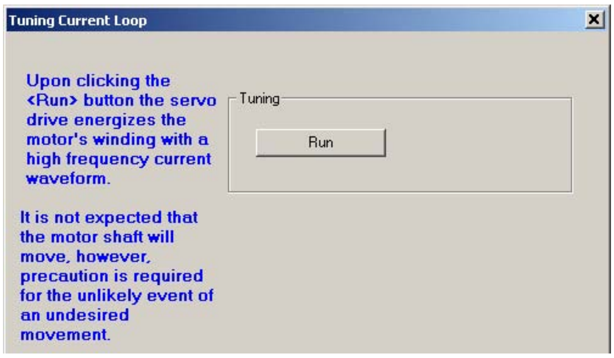
You may stop at any step in the process and the Wizard will save all the parameters in the drive memory.

8. To continue with initial drive setup, select either **Step 1** (for digital brushless motors) or **Step 3** (for rotating brush motors).

The following sections describe each tuning step in turn.

2.7 Tuning the Current Loop

The Composer Wizard begins the tuning process with the current loop because the current controller must be properly tuned before any other process can be successfully carried out. When you select Step 1, **Tuning Current Loop** from the Custom dialog box, a Tuning Current Loop dialog box similar to the following is displayed.



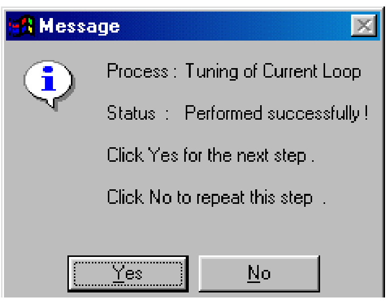
This step energizes the motor winding with a high-frequency current, in order to identify the dynamic response for resistance and inductance. The result of the test is a set of well-tuned current controller parameters. The controller performance is verified by a test graph showing the Reference, Response and Controller Out vectors. The controller output is proportional to the motor voltage and is shown in order to verify that during the test, the motor voltage did not saturate. If the motor voltage *does* saturate (is clipped at the top), the response displayed will not reflect the dynamics of the current control loop but rather the dI/dT limits of the power supply and inductance. In such a case, use lower Continuous Current levels (defined in the System Definitions and Limits dialog box, section ☐) and repeat the current tuning process from the beginning.

To tune the current loop:

1. Click **Run** in the dialog box. The current will flow into the motor A phase and divide equally between the B and C phases. The motor current generates a fixed magnetic field, so that the motor will jump in the direction of the magnetic field and remain there.

Be aware that the motor shaft may move while the current loop is being tuned, up to half an electrical rotation at most. Therefore, take the necessary precautions for the unlikely event of an undesired movement.

Upon completing the test, the system will store the current tuning information and display the following message:



2. Click **Yes** to continue directly to the next tuning step.

2.8 Configuring Commutation

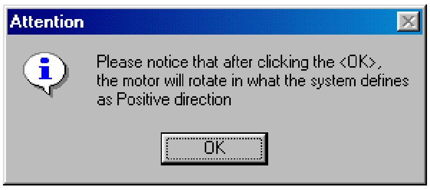
For all brushless motors, you need to define the parameters for commutation between the servo drive and the motor. This process automatically determines the correct phase sequence needed for optimal commutation. When digital Hall sensors are used, the system also finds the offset from the commutating points.

The Composer performs commutation setup by driving the motor in stepper mode in both directions; you may select positive or negative counts for each physical direction. The configuration process relies on the parameters entered previously in the Commutation Feedback Parameters dialog box (section 2.4).

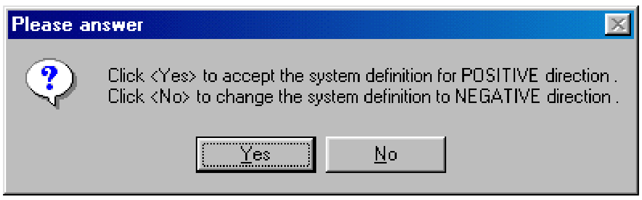
The Composer Wizard displays the Establishing Commutation dialog box when you select Step 2, **Establishing Commutation** from the Custom dialog box, or when you click **Next** in the Tuning Current Loop dialog box.

To perform commutation setup:

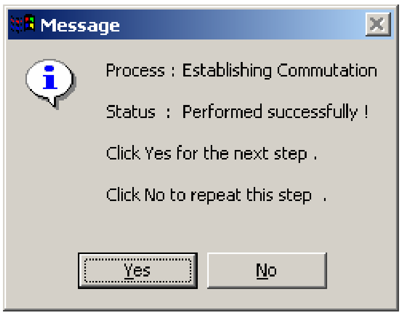
1. Click **Run**. The motor shaft will move until the Composer finds all the needed commutation parameters. The following message will be displayed:



2. Click **OK**. The drive will rotate the motor in what the system assumes is the positive direction. After the rotation is complete, the following message will be displayed:



3. In order to define the positive and negative directions for the motor, click either **Yes** to accept the present motor direction as positive or click **No** to change the present motor direction to negative. The motor will then move back to its starting position (for subsequent tuning) and the system will store the commutation information. The following message will be displayed:



4. Click **Yes** to complete the commutation setup and to continue to the next tuning step.

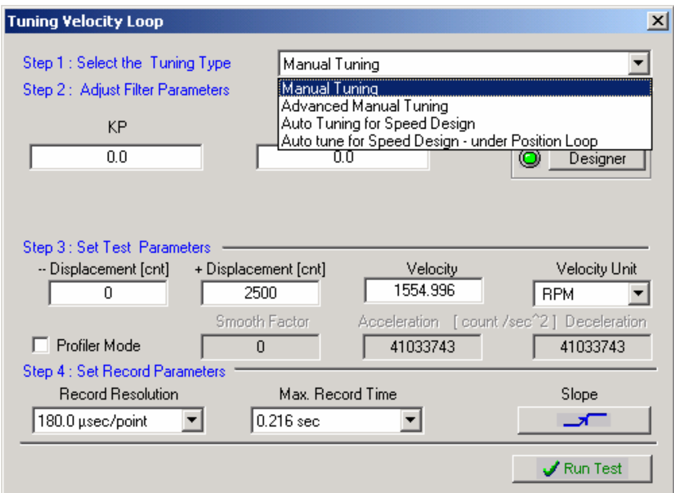
2.9 Tuning the Velocity Loop

This process enables you to tune the velocity loop and to set an optimal balance between control gains and precise motion on the one hand; and higher stress, measurement and quantization noise on the other.

The Composer Wizard enables you to perform this process in three modes:

- Manual tuning, in which you manually configure the velocity loop by entering each of the required parameters.
- Automatic tuning, in which you select the level of fine tuning, and let the Wizard auto-tuner determine the needed parameters automatically.
- Advanced manual tuning, in which you create or modify a gain schedule table to operate with a range of velocities.
- Auto tune for Speed Design – under Position Loop, ________________________ ____.

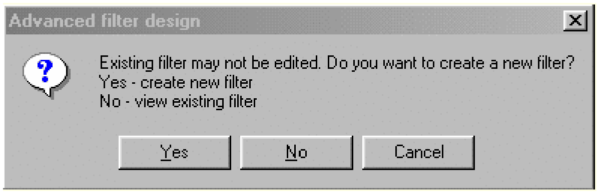
When you select Step 3, **Tuning Velocity Loop** in the Custom dialog box or when you finish the Establish Commutation step, the Tuning Velocity Loop dialog box is displayed as follows:



You may tune the velocity loop manually (section 2.9.1), have the Wizard perform automatic tuning (section 2.9.2) or perform advanced manual tuning (section 2.9.3).

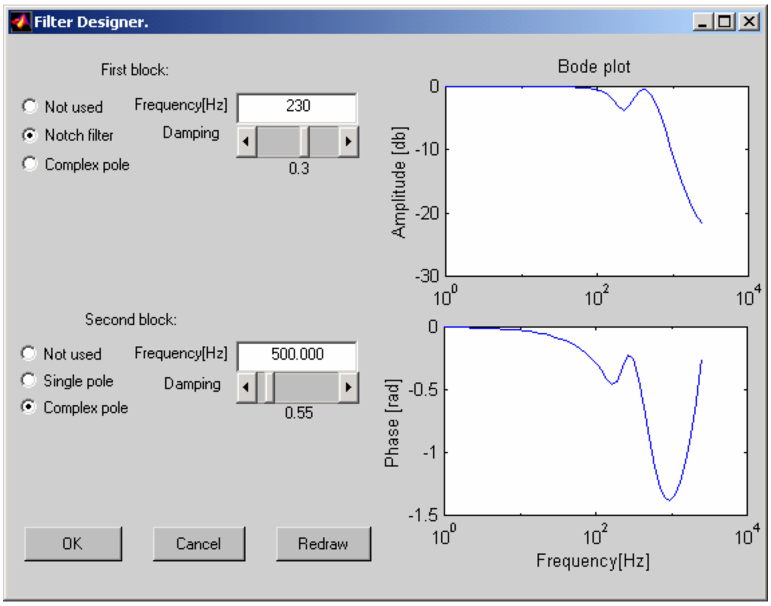
2.9.1 Manually Tuning the Velocity Loop

1. From the **Step 1: Select the Tuning Type** drop-down list, select **Manual Tuning**. The dialog box will remain with its default settings for manual tuning.
2. In the **KP** and **KI** text boxes, type the required proportional gain and integral gain filter values for optimal step response.
3. In certain cases, you may wish to manually fine-tune the frequency and resonance definitions by designing a custom filter. To do so, click the **Designer** button in the **Advanced Filter** block. If the existing filter cannot be edited (because it has been created by the auto-tuner or if the system sampling time has changed from the last filter modification), the following message will be displayed:



Click **Yes** to open the Filter Designer in order to create a new filter, or **No** to view the parameters and Bode plot of the existing filter.

The Filter Designer dialog box will be displayed as follows:



This dialog box enables you to select a filter component and — using the Bode plots — determine how it may reduce resonance at selected frequencies. Each filter consists of two components: **Second order component** and **Pole component**. You may define each component or disable it by selecting **None**. Full details for designing a new filter are given in the Appendix to this manual.

4. In the **Step 3: Set Test Parameters** block of the Tuning Velocity Loop dialog box, enter the **Displacement** and **Velocity** values (selecting the **Velocity Unit** that is convenient to you). Together, these settings define the step command to be used during testing. The **-Displacement** and **+Displacement** values (in encoder units) indicate the upper and lower limits in which the system moves during the test phase. The default value is one revolution divided by the number of electric poles in a revolution.
5. You may use the Profiler (select the **Profiler Mode** checkbox) to profile a response according to values that you set. In Profiler mode, you define the velocity command according to **Smooth Factor**, **Acceleration** and **Deceleration**. Smooth factor is used to "soften the sharp corners" of the motion speed profile, according to the acceleration and deceleration values that you define. Smoothing a profile increases the time required to complete the motion, due to higher acceleration and deceleration times.
6. In **Step 4: Set Record Parameters**, select the desired recording parameters as follows:
   - **Record Resolution** (μsec) is the test data record resolution.
   - **Maximum Record Time** (sec) is the test data record time length.
7. Click **Run Test** to begin the manual velocity loop tuning. The motor will begin to move back and forth, and the system will record the step response. Upon completion, the motor will be disabled and a step response graph will be displayed, as in the following example:

Figure 2-1: ***Step Response Graph - Manual Velocity Loop Tuning***

8. Evaluate the results in the graph and repeat steps 2 to 6 as necessary.
9. Click **Run Test** to repeat the tuning process and to produce an updated graph.
10. Repeats steps 8 and 9 until your graph indicates optimal tuning.

2.9.2 Automatically Tuning the Velocity Loop

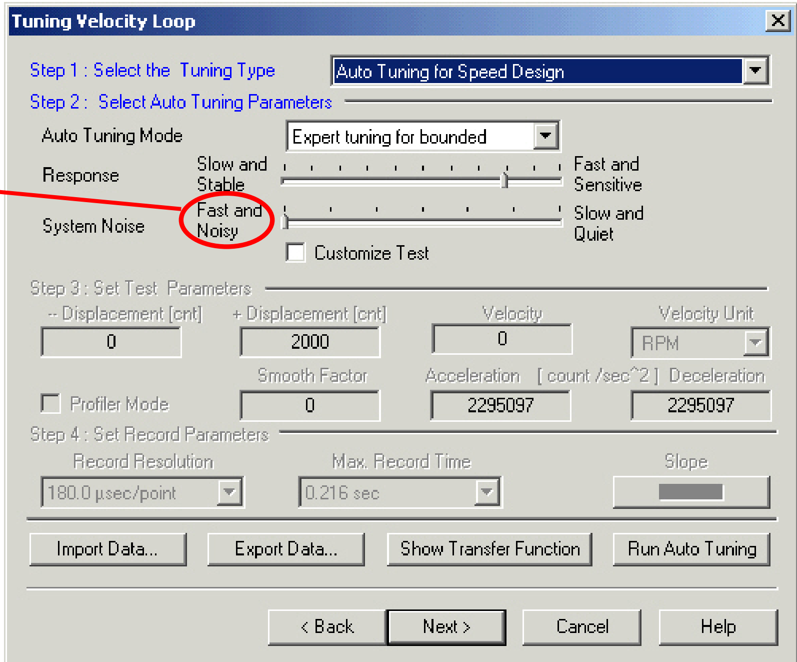
Using the auto-tuning mode for this process provides more precise velocity loop tuning.

To set the auto-tuning options:

1. From the **Step 1: Select the Tuning Type** drop-down list, select one of the Auto Tuning type options:
   - **Auto Tuning for Speed Design**: for a stand-alone speed mode, in which the drive works only with the velocity loop.
   - **Auto Tuning of Speed Design – Under Position Loop**: when the drive receives speed commands from the position controller. When selecting this option, be sure that the position controller is not active during the tuning.

   When you make your auto-tune selection, the text boxes of the data box will change accordingly.

Tuning may be noisy



2. In the **Step 2: Select Auto Tuning Parameters** block, select the tuning mode from the **Auto Tuning Mode** drop-down list:
   - **Expert tuning for bounded motion** (default): In this mode, the motor swings around a chosen fixed point. Use this option for linear motors and in cases where the motor shaft must remain within position boundaries. In this mode, the boundaries are designated by an algorithm based on system settings; they are displayed before the tuning process begins. If the motion is *not* restricted, the **Expert tuning for free motion** option yields even better results.
   - **Expert tuning for free motion**: In this mode, the motor rotates freely and is therefore not applicable to linear motors. This option is recommended if there are no restrictions on motion (angle and position); it gives more precise results.

3. Use the **Response** slider to select the system margin you require:
   - Choose a value towards **Fast and Sensitive** if you require a more responsive (agile) system.
   - Choose a value towards **Slow and Stable** if you need a more robust system, which is required in a number of cases, among them:
     - Machines whose mechanics may vary, as when handling loads of different masses
     - A machine type that uses one type of control parameters, although the mechanics of each machine may differ from each other.
4. Use the **System Noise** slider to set the plant noise level: a value towards **Slow and Quiet** reduces the system noise more effectively. It is recommended to start with an average level and then observe the step response at the plant input. The current is built from two signals: a part that varies slowly due to the reference command and a signal that varies quickly due to system noise. If the noise saturates the plant or its fast-varying signal at the plant output, adjust the **System Noise** level accordingly. Then run the test again.
5. To manually select the test parameters — rather than having the Wizard calculate them for you — click the **Customize Test** checkbox, and in the **Displacement** and **Velocity** text boxes, enter the values that define the step command to be used for testing:
   - **-Displacement** and **+Displacement** are the upper and lower limits in which the system may move. The default value for automatic mode, for both parameters, is one revolution divided by the number of electric poles in a revolution.
   - **Velocity** is the square wave speed reference amplitude command.
6. You may use the Profiler (select the **Profiler Mode** checkbox) to profile a response according to values that you set. In Profiler mode, you define the velocity command according to **Smooth Factor**, **Acceleration** and **Deceleration**. The smooth factor is used to "soften the sharp corners" of the motion speed profile, according to acceleration and deceleration values that you define. Smoothing a profile increases the time required to complete the motion, due to higher acceleration and deceleration times.
7. In the **Step 4: Set Record Parameters** block, select the desired recording parameters:
   - **Record Resolution** (µsec) is the test data record resolution.
   - **Maximum Record Time** (sec) is the test data record time length.
8. Click **Run Auto Tuning** to start the auto-tuning design process. If the automatic test parameters have been selected (the **Customize Test** checkbox is not selected), the Wizard will automatically set the test parameters according to the current, speed and displacement limitations of the controller, giving what should be the optimal results. During the test, the motor will move backwards and forward, recording the step response. The results will be displayed in a step response graph comparing the velocity and the current command, as in the following example:

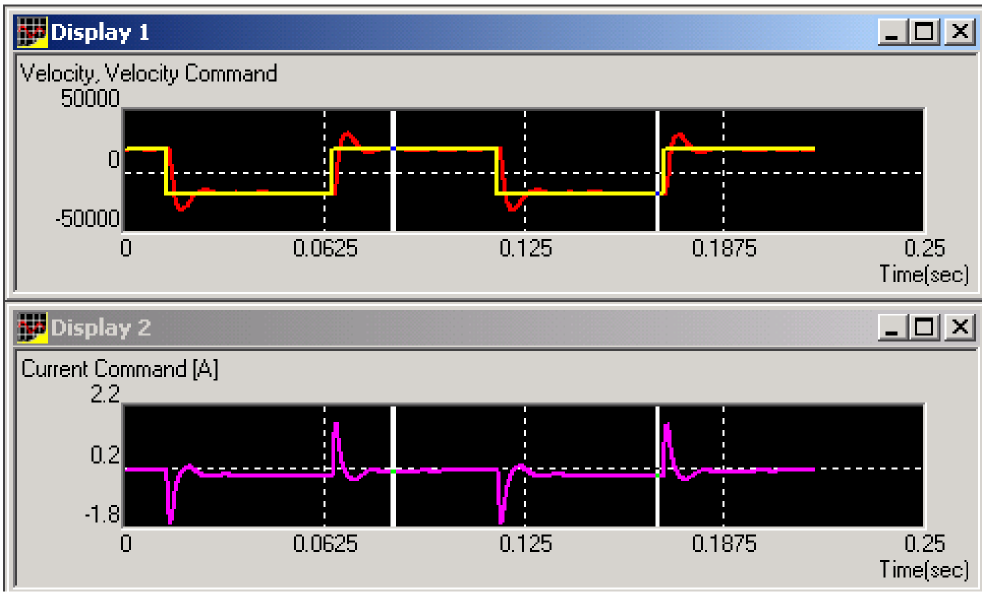


Figure 2-2: ***Step Response Graph - Automatic Velocity Loop Tuning***

9. After the first test, check the current response at the plant input. The current is built from two signals: one that slowly varies due to the reference command, and one that changes quickly due to system noise. When analyzing the graph, estimate the contribution of the system noise; if the noise saturates the plant or if the fast-varying signal at the plant output is not acceptable, set the **System Noise** slider to **Slow and Quiet**. However, be aware that this may decrease the system performance to some extent. You may also wish to decrease the reaction time; in this case, set the slider towards **Fast and Noisy**. When you have made your adjustments, click **Run** to apply the new values, recalculate the controller parameters and produce a new step response.
10. You may further manipulate the data by using the following command buttons:
    - **Export data**: Saves the identification data in case of fault or if errors occur during the auto-tuning process. This is useful when troubleshooting with Elmo support personnel.
    - **Import data**: Restores the identification data from files saved in the Export Data process. In this case, the system will execute the final test without undergoing the extended identification process.
    - **Show transfer function**: Displays a bode plot of the system transfer functions for open and closed loops.
11. When you are satisfied that you have achieved optimal velocity loop tuning, you can save your tuning results (recommended). To do so, click **Export Data**, enter a name for the file, browse to the save location and click **Save**.
12. Click **Next** to continue to the next step, tuning the position or dual loop.

2.9.3 Performing Advanced Manual Tuning - Velocity Loop

Reviewing the two previous methods of velocity tuning: *Manual* tuning enables you to manually determine optimal gain filter values for a given velocity, while *automatic* tuning can provide fairly precise velocity tuning parameters in a relatively short time. In cases where precise gain values are needed for a range of velocities, the **Advanced Manual Tuning** option may be desired instead.

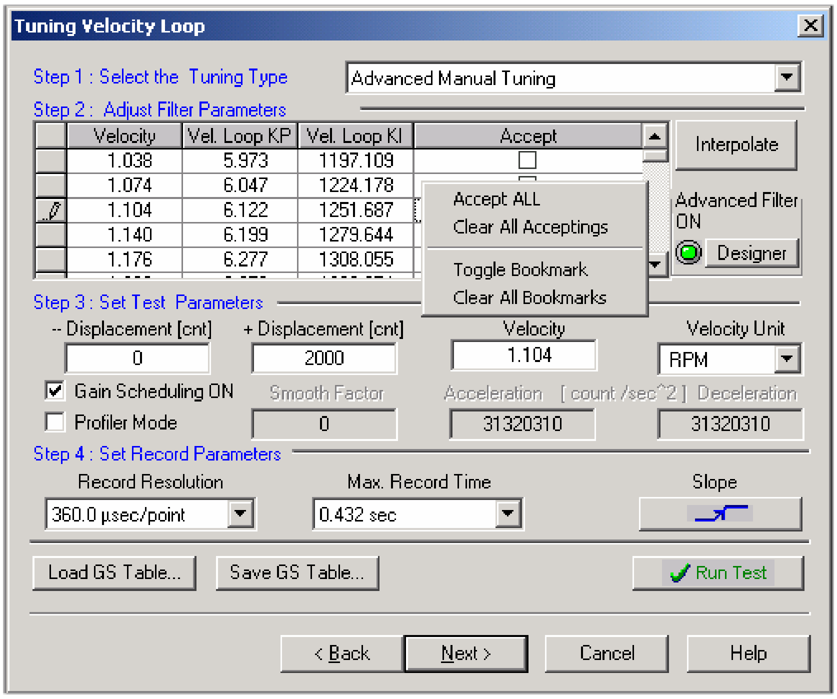
The advanced manual tuning procedure gives you full control over gain scheduling, enabling you to create or edit a gain schedule table for 64 different velocity values. You may modify a table created previously (through auto-tuning or this option) or you may build an entirely new table.

When performing advanced manual tuning, the Composer Wizard gets you the option to determine optimal values for some or all of the 64 entries. You may determine the optimal gains for selected velocities and then have the Wizard perform interpolation to update the rest of the list according to those values. When you are satisfied with the entire table, you can save it for on-going use.

To perform advanced manual tuning:

1. From the **Step 1: Select the Tuning Type** drop-down list of the Tuning Velocity Loop dialog box, select **Advanced Manual Tuning**. The dialog box will be redisplayed.

   If the connected drive has been tuned previously, the existing gain schedule table will be displayed; otherwise, the **Velocity** column will contain the values defined according to the sample times of the specific connected drive, with zeroes for KP and KI values.



> **Note:**
>
> To use a gain schedule table created through auto-tuning or from a previous advanced manual tuning sessions, click **Load GS Table** at the bottom and select the desired file from the Open dialog box.

2. The recommended procedure for determining the appropriate gain values for *a specific velocity* is as follows:
   a. Be sure that the **Gain Scheduling ON** check box (in the middle) is not selected and that the Advanced Filter Designer is **OFF**. Select a velocity value from the list or enter a new velocity value in a row appropriate to its place in the velocity range.
   b. Enter values for the **Vel. Loop KP** (proportional gain) and **Vel. Loop KI** (integral gain) coefficients.
   c. In the **-Displacement** and **+Displacement** text boxes, be sure that the values are relevant for the selected velocity. Change them as needed.
   d. In the **Step 4: Set Record Parameters** block, select the desired recording parameters:
      - **Record Resolution** (μsec) is the test data record resolution.
      - **Maximum Record Time** (sec) is the test data record time length.
   e. Click **Run Test**. The system will send the gain coefficients for the selected row to the drive and the motor will operate at the selected velocity, turning according to the **-Displacement** and **+Displacement** values. The recorder will automatically record the feedback, command and current, and display the results for you to determine if the values you entered are appropriate.
   f. If needed, modify your values, repeating steps b. through f. until you are satisfied that the gain values for that velocity are acceptable. You may then highlight the row to indicate that it has been tested satisfactorily. To do so, click the right mouse button and from the shortcut menu, select **Toggle Bookmark**. The row with the bookmark will be highlighted in a light color.
   g. Repeat steps b. through g. for one or more other velocity values until you are satisfied that you have determined a representative amount of recommended KI and KP values.

3. You may have the Wizard complete the gain schedule table for you and then *test the entire table with gain scheduling.* To do so, perform the following procedure:
   a. Check the **Gain Scheduling ON** check box.
   b. In the row indicating the lowest velocity value for the range to be filled in, and in the row indicating the highest value in the range, select **Accept**.
   c. Click **Interpolate**. The Wizard will calculate KP and KI values for all velocities in the range between the lowest row in which you checked **Accept** and the highest row in which you checked **Accept**. Rows before and after that range will take the values of the first and last **Accept**ed row, respectively.

d. To test the interpolated velocity values, select a row of the table that indicates the velocity to be reached during the test. Alternatively, in the **Velocity** text box, enter the velocity value. If that exact value does not appear in the table, the value closest to it — but not exceeding it — will be selected in the table.

e. Enter the **-Displacement** and **+Displacement** values to be used for the test.

f. In the **Step 4: Set Record Parameters** block, select the desired recording parameters:
   - **Record Resolution** (μsec) is the test data record resolution.
   - **Maximum Record Time** (sec) is the test data record time length.

g. Click **Run Test.** The system will send the gain coefficients to the drive and activate the gain scheduling. During the test, the motor will turn according to the -Displacement, +Displacement and other profile parameters. The recorder will be activated automatically in order to record and display the results of the speed, speed command and current command.

Tips:

- You can remove bookmarks using the **Clear All Bookmarks** option from the right-click shortcut menu.
- You can also use the shortcut menu to manipulate the list of rows that you have accepted. To automatically accept all rows, click the right mouse button and then select **Accept ALL** to mark all rows as **Accept**ed. (If all rows are accepted, you may use the opposite command — **Clear All Accepts** — to remove checks from all **Accept** check boxes.)

4. To modify the gain schedule table further, repeat steps 2 and 3 as necessary.

5. To save your final gain schedule table, click **Save GS Table**. In the Save As dialog box, enter a name for the file, browse to the location at which it should be saved, and click **Save**. The system will save the table, along with its bookmarks.

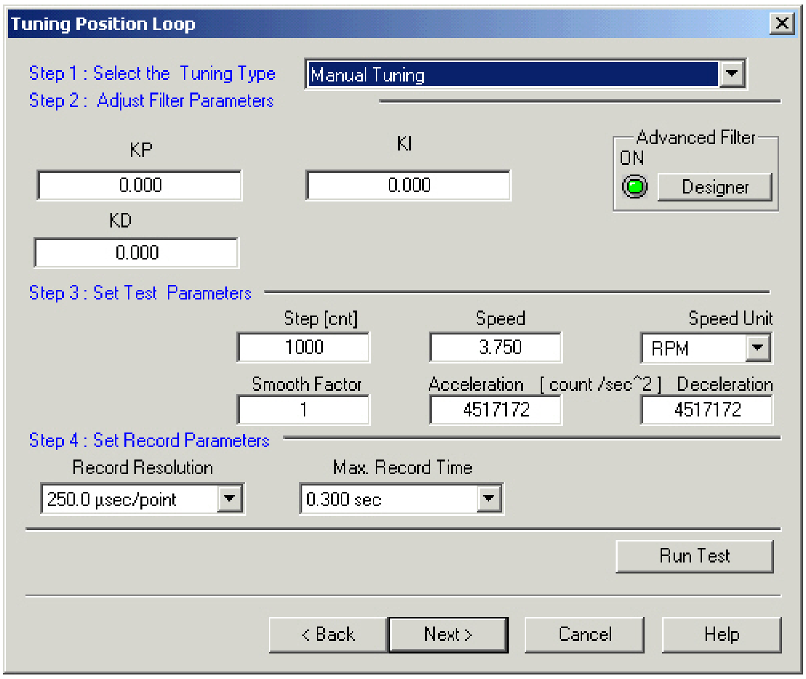
2.10 Tuning the Position Loop

According to your drive type, you will now need to tune either the position loop or the dual — position/velocity — loop. Tuning of the dual loop is described in section 2.11. The procedure is very similar for both loops. The Wizard performs the test using point-to-point (PTP) motions, according to the selected PTP motion and recording parameters. The Composer records the PTP performance and compares it to the PTP trajectory.

The Composer Wizard enables you to perform position loop tuning in three modes:

- Manual tuning, in which you manually configure the position loop by entering each of the required parameters.
- Automatic tuning, in which you select the level of fine tuning, and let the Composer determine the needed parameters automatically.
- Advanced manual tuning, in which you create or modify a gain schedule table that operates for a range of velocities.

When you select Step 4, **Tuning Position Loop** in the Establishing Commutation dialog box or when you finish the Tuning Velocity Loop process, the Tuning Position Loop dialog box is displayed as follows:



Select the tuning mode from the **Step 1: Select Tuning Type** drop down list.

2.10.1 Manually Tuning the Position Loop

1. From the **Step 1: Select the Tuning Type** drop-down list, select **Manual Tuning**. The dialog box will remain with its default settings.
2. In the **Step 2: Adjust Filter Parameters** block, type the filter values required for optimal step response in the **KP** (proportional gain), **KI** (integral gain) and **KD** (derivative gain) text boxes.

You may click the **Designer** button in the **Advanced Filter block** at the right to better define the filter values. Refer to section 2.9.1 for general instructions, and to the Appendix for detailed explanations about using the Filter Designer.

3. In the **Step 3: Set Test Parameters** block, enter the parameters needed to define the motor position. These values define the step command to be used for testing.
   - **Step**: the level at which the step command begins, at the system location prior to start of the test.
   - **Speed**: the maximum speed allowed for the PTP motion. This value may not be reached if the motion distance and the allowed accelerations are small.
   - **Smooth Factor**: the time required to reach full acceleration.
   - **Acceleration**: the rate at which the profiled motion gains speed.
   - **Deceleration**: the rate at which the profiled motion slows down.
4. In **Step 4: Set Record Parameters**, select the desired recording parameters:
   - **Record Resolution** (µsec) is the test data record resolution.
   - **Maximum Record Time** (sec) is the test data record time length.
5. Click **Run Test** to begin the position loop tuning. The motor will begin to move back and forth and the system will record the step response. Upon completion, a step response graph will be displayed as in the following example:

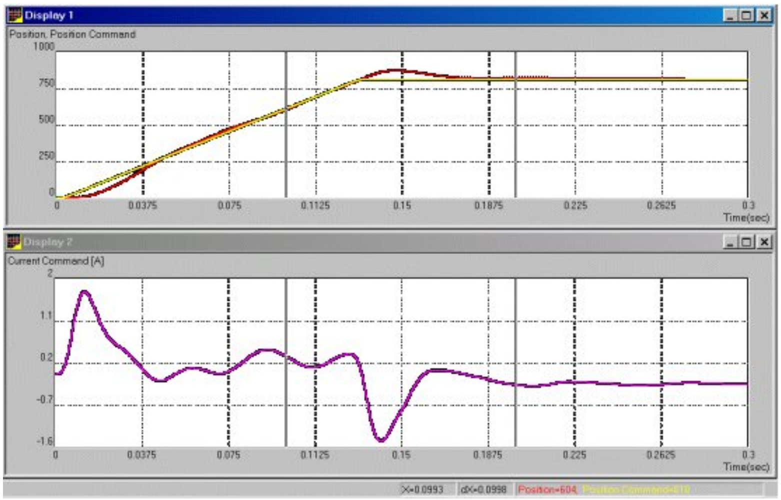


*Figure 2-3: **Step Response Graph - Manual Position Loop Tuning***

6. Evaluate the results in the graph and repeat steps 2 to 5 as necessary.
7. Click **Run Test** to repeat the tuning process and to produce an updated graph.
8. Repeats steps 6 and 7 until your graph indicates optimal tuning.

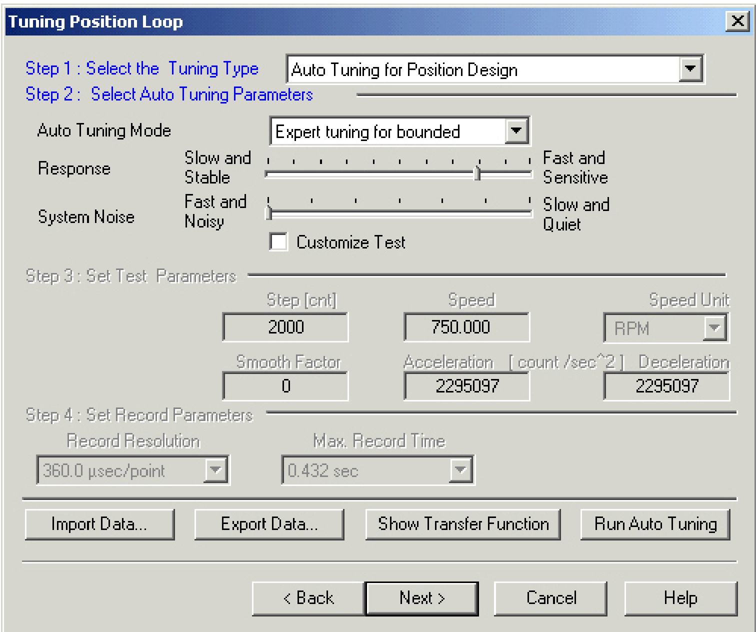
2.10.2 Automatically Tuning the Position Loop

Using the auto-tuning mode provides more precise position loop tuning. This process is almost identical with the auto-tuning process for the velocity loop.

To set the auto-tuning options:

1. From the **Step 1: Select the Tuning Type** drop-down list, select **Auto Tuning for Position Design**.

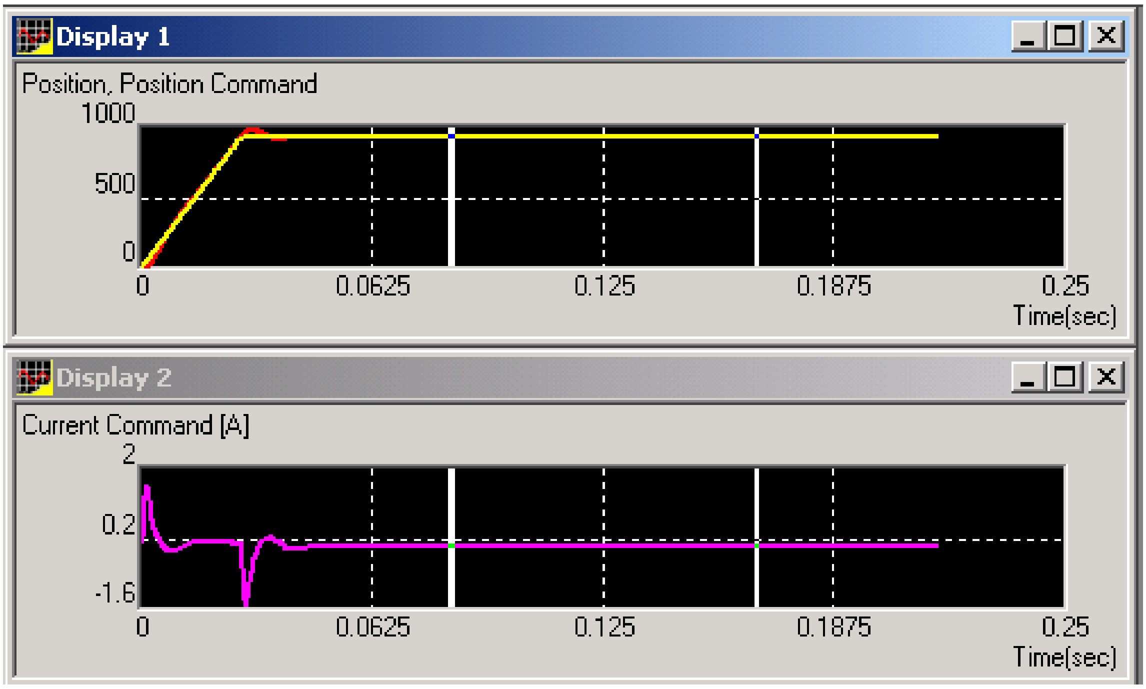
   When you make your auto-tune selection, the text boxes of the data box will change accordingly.



2. In the **Step 2: Select Auto Tuning Parameters** block, select the tuning mode from the **Auto Tuning Mode** drop-down list:
   - **Fast tuning**: This mode is applicable for simple systems, where less precise results are adequate. It can be used if there are no restrictions on the motion boundaries, and if you prefer to minimize the time and power required by the system during the identification stage. You should use this mode only if the duration of the identification phase prevents the use of an Expert mode.
   - **Expert tuning for bounded motion** (default): In this mode, the motor swings around a chosen fixed point. Use this option if the motor shaft must remain within position boundaries. The boundaries are designated by an algorithm based on system settings; they are displayed before the tuning process begins. If the boundaries are beyond the physical possibilities of the application, the Wizard will switch to manual tuning.
   - **Expert tuning for free motion**: In this mode, the motor rotates freely and is therefore not applicable to linear motors. This option is recommended if there are no restrictions on the system motion (angle and position); it gives more precise results.

3. Use the **Response** slider to select the system performance margin that you require and set the **System Noise** slider to the optimal plant noise level (refer to section 2.9.2).

4. In the middle of the dialog box, the Wizard displays the default test parameters. To manually define these values, click **Custom Test** and enter the following values in the **Step 3: Set Test Parameters** and **Step 4: Set Record Parameters** blocks:
   - Step: the level at which the step command begins, at the system location prior to the start of the test.
   - Speed: the maximum speed allowed for the PTP motion. This value may not be reached if the motion distance and the allowed accelerations are small.
   - Smooth Factor: the time required to reach full acceleration.
   - Acceleration: the rate at which the profiled motion gains speed.
   - Deceleration: the rate at which the profiled motion slows down.
   - Record Resolution (µsec) is the test data record resolution.
   - Maximum Record Time (sec) is the test data record time length.
5. Click **Run Auto Tuning** to start the auto-tuning process. Review the resulting graph and revise your values (repeating steps 3 and 4) as needed until your results are satisfactory.
6. When you are satisfied that you have achieved optimal position loop tuning, you can save your tuning results (recommended). To do so, click **Export Data**, enter a name for the file, browse to the save location and click **Save**.
7. Click **Next** to continue to the next step, saving your application setup data.

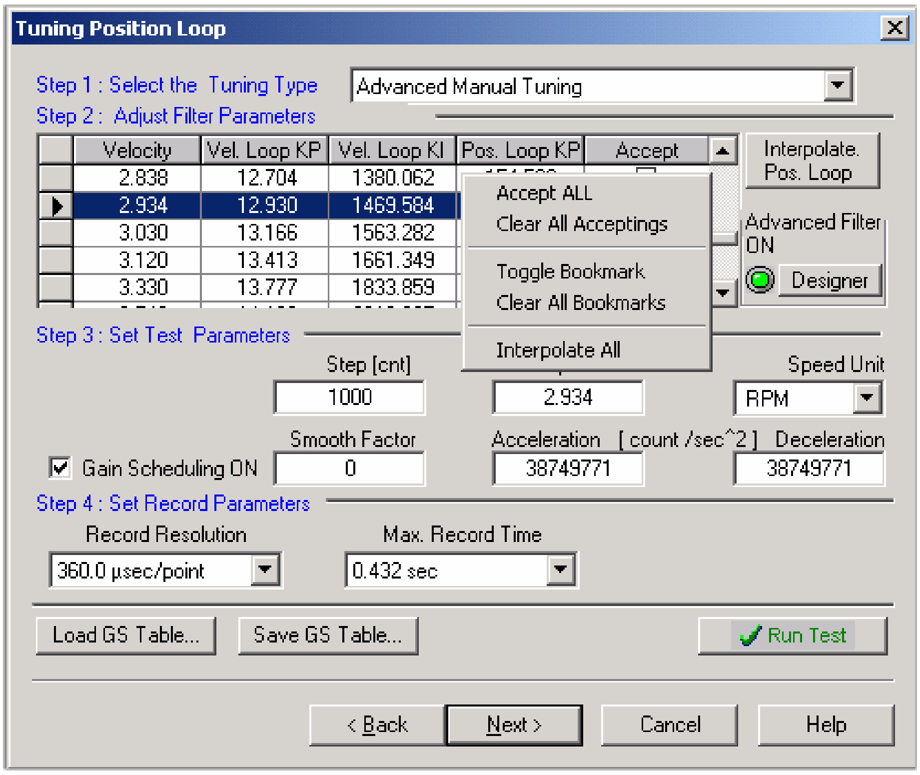


2.10.3 Performing Advanced Manual Tuning - Position Loop

As explained in section 2.9.3, the advanced manual tuning procedure can be used to manually define the gain schedule table. You may define any number of gain scheduling values and have the Wizard interpolate the entire list. For positioning loop tuning, you may use a list previously defined during velocity loop tuning.

To perform advanced manual tuning for the position loop:

1. From the **Step 1: Select the Tuning Type** drop-down list of the Tuning Position Loop dialog box, select **Advanced Manual Tuning**. The dialog box will be redisplayed as follows:



Because a PI controller is used for velocity, and a different — PID.— controller is used for position, the **KP**, **KI** and **KD** values will need to be defined for this gain schedule, even if a gain schedule was created previously during velocity tuning.

> **Note:**
> To use a gain schedule table created previously, click **Load GS Table** at the bottom and select the desired file from the Open dialog box.

2. The recommended procedure for determining the appropriate gain values for *a specific velocity* is as follows:

   a. Be sure that the **Gain Scheduling ON** check box (in the middle) is not selected and that the Advanced Filter Designer is **OFF**.

   c. Select an existing velocity or enter a new velocity value in a row appropriate to its place in the velocity range.

d. Enter the relevant gain coefficients for that velocity.

e. In the **Step 3: Set Test Parameters** and **Step 4: Set Record Parameters** blocks, be sure that the required values are entered.

e. Click **Run Test**. The system will send the gain coefficients for the selected row to the drive and the motor will operate at the selected velocity and according to the test and record parameters that you selected. The recorder will automatically record the feedback and display the results for the highlighted row.

f. Evaluate the results and, if needed, modify your values, repeating steps b. through e. until you are satisfied that the gain values for that velocity are acceptable. You may then highlight the row to indicate that it has been tested satisfactorily. To do so, click the right mouse button and from the shortcut menu, select **Toggle Bookmark**. The row with the bookmark will be highlighted in a light color.

g. Repeat steps b. through f. for one or more other velocity values until you are satisfied that you have determined a representative amount of recommended gain scheduling values.

3. You may have the Wizard complete the gain schedule table for you and then *test the entire table with gain scheduling.* To do so, perform the following procedure:

a. Check the **Gain Scheduling ON** check box.

b. In the row indicating the lowest velocity value for the range to be filled in, and in the row indicating the highest value in the range, select **Accept**.

c. To have the Wizard "complete" the gain scheduling for that range, click the right mouse button and from the shortcut menu, click **Interpolate Pos. Loop**.

The Wizard will calculate the coefficient values in the range you defined in step b. Rows before and after that range will take the values of the first and last **Accept**ed row, respectively.

d. To test the interpolated values, select a row of the table that indicates the velocity to be reached during the test. Alternatively, in the **Speed** text box, enter the velocity value. If that exact value does not appear in the table, the value closest to it — but not exceeding it — will be selected in the table.

e. In the **Step 3: Set Test Parameters** and **Step 4: Set Record Parameters** blocks, be sure that the required values are entered.

f. Click **Run Test.** The system will send the gain coefficients to the drive and activate the gain scheduling. During the test, the motor will turn according to the test and recording parameters. The recorder will be activated immediately in order to record and display the results of the position, position command and current command.

	Remember: ▪ You can remove bookmarks using the **Clear All Bookmarks** option from the right-click shortcut menu. ▪ You can also automatically accept all rows of the table by selecting the **Accept ALL** option from the shortcut menu. (Use **Clear All Accepts** to remove all checks.)

4. To modify the gain schedule table further, repeat steps 2 and 3 as necessary.

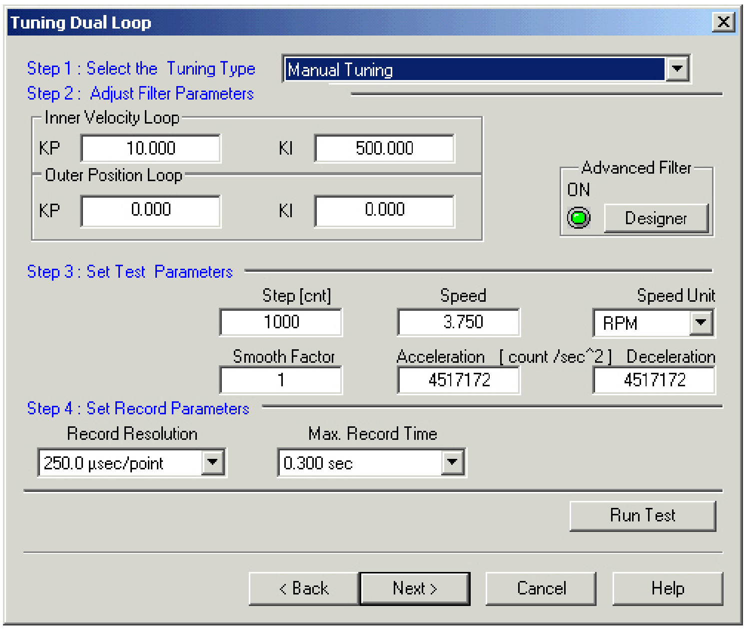
5. To save your final gain schedule table, click **Save GS Table**. In the Save As dialog box, enter a name for the file, browse to the location at which it should be saved, and click **Save**. The system will save the table, along with its bookmarks.

2.11 Tuning the Dual Loop

Tuning a dual loop is very similar to tuning a position loop. As with the position loop, the Wizard performs the test using point-to-point (PTP) motions and the Composer records the PTP performance and compares it to the PTP trajectory. Like the velocity and position loops, you have three options:

- Manual tuning, in which you manually configure the position loop by entering each of the required parameters.
- Automatic tuning, in which you select the level of fine tuning, and let the Composer determine the needed parameters automatically.
- Advanced manual tuning, in which you create or modify a gain schedule table that operates for a range of velocities.

When you select Step 5, **Tuning Dual Loop** in the Establishing Commutation dialog box or when you finish the Tuning Velocity Loop process, the Tuning Dual Loop dialog box is displayed as follows:



Select the tuning mode from the **Step 1: Select Tuning Type** drop down list.

2.11.1 Manually Tuning the Dual Loop

1. From the **Step 1: Select the Tuning Type** drop-down list, be sure that **Manual Tuning** is selected. The dialog box will remain with its default settings.

2. In the **Step 2: Adjust Filter Parameters** block, type the filter values required for optimal step response in the **KP** and **KI** (proportional gain and integral gain) text boxes of both the the **Inner Velocity Loop** and **Output Position Loop** blocks.

   You may click the **Designer** button in the **Advanced Filter** block to better define the filter values. Refer to section 2.9.1 for general instructions, and to the Appendix for detailed explanations about using the Filter Designer.

3. In the **Step 3: Set Test Parameters** block, enter the parameters needed to define the motor position. These values defines the step command to be used for testing.
   - **Step**: the level at which the step command begins, at the system location prior to start of the test.
   - **Speed**: the maximum speed allowed for the PTP motion. This value may not be reached if the motion distance and the allowed accelerations are small.
   - **Smooth Factor**: the time required to reach full acceleration.
   - **Acceleration**: the rate at which the profiled motion gains speed.
   - **Deceleration**: the rate at which the profiled motion slows down.

4. In **Step 4: Set Record Parameters**, select the desired recording parameters:
   - **Record Resolution** (μsec) is the test data record resolution.
   - **Maximum Record Time** (sec) is the test data record time length.

5. Click **Run Test** to begin the dual loop tuning. The motor will begin to move back and forth and the system will record the step response. Upon completion, a step response graph will be displayed for evaluation.

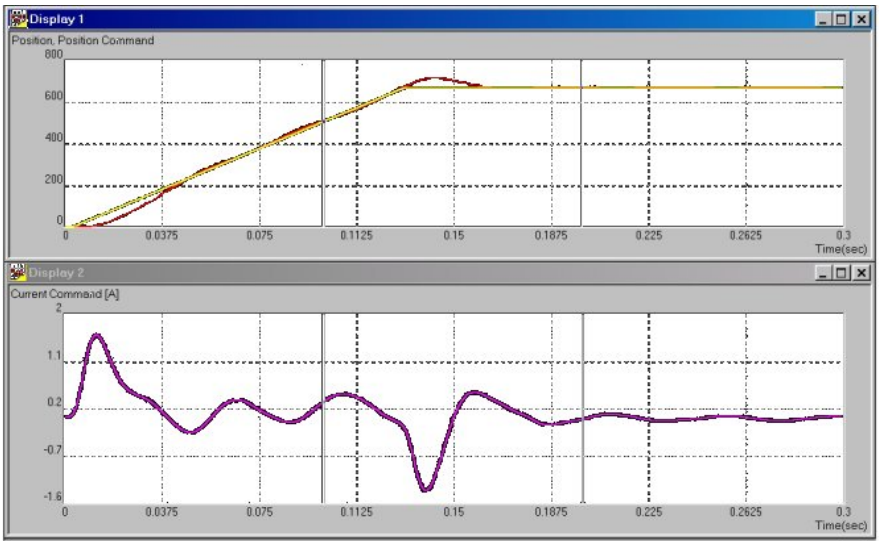


*Figure 2-4: **Step Response Graph - Dual Loop Tuning***

6. As in previous tests, revise your values as necessary and rerun the tests until the results are satisfactory.

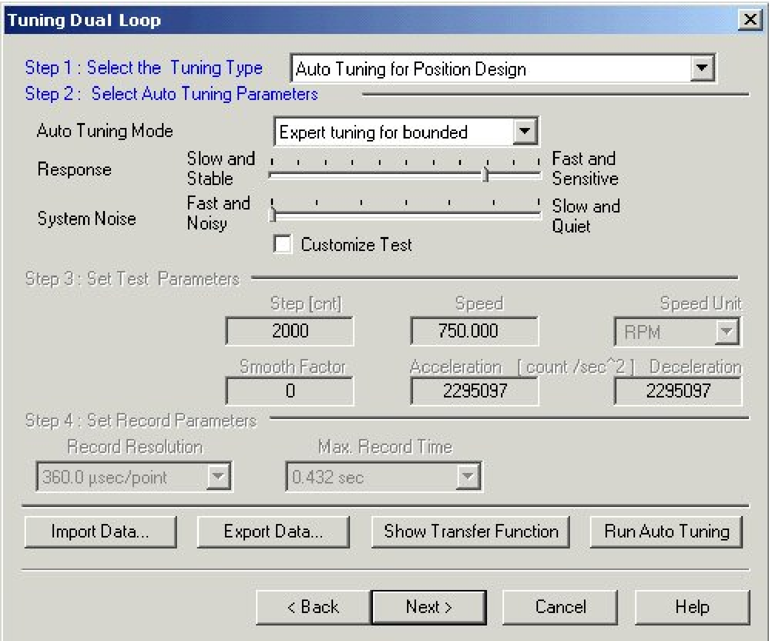
2.11.2 Automatically Tuning the Dual Loop

Using the auto-tuning mode provides more precise position loop tuning. This process is almost identical with the auto-tuning process for the velocity loop.

Set the auto-tuning options as you did in the Tuning Velocity Loop dialog box:

1. From the **Step 1: Select the Tuning Type** drop-down list, select **Auto Tuning for Dual Loop Design**.

   When you make your selection, the text boxes of the data box will change accordingly.



2. In the **Step 2: Select Auto Tuning Parameters** block, the **Expert tuning for bounded motion** option will be selected; it is the only option available for dual loops.

3. Use the **Response** slider to select the system performance margin that you require and set the **System Noise** slider to the optimal plant noise level (refer to section 2.9.2).

4. In the middle of the dialog box, the Wizard displays the default test parameters. To manually define these values, click **Custom Test** and enter the following values in the **Step 3: Set Test Parameters** and **Step 4: Set Record Parameters** blocks:
   - **Step**: the level at which the step command begins, at the system location prior to the start of the test.
   - **Speed**: the maximum speed allowed for the PTP motion. This value may not be reached if the motion distance and the allowed accelerations are small.
   - **Smooth Factor**: the time required to reach full acceleration.
   - **Acceleration**: the rate at which the profiled motion gains speed.
   - **Deceleration**: the rate at which the profiled motion slows down.
   - **Record Resolution** (µsec) is the test data record resolution.
   - **Maximum Record Time** (sec) is the test data record time length.

5. Click **Run Auto Tuning** to start the auto-tuning process. Review the resulting graph and revise your values as needed until your results are satisfactory.

6. When you are satisfied that you have achieved optimal dual loop tuning, you can save your tuning results (recommended). To do so, click **Export Data**, enter a name for the file, browse to the save location and click **Save**.

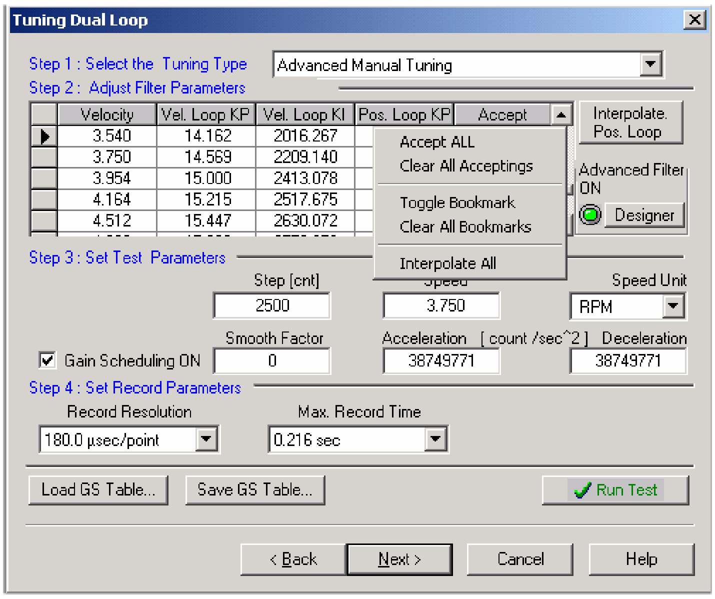
7. Click **Next** to continue to the next step, saving your application setup data.

2.11.3 Performing Advanced Manual Tuning - Dual Loop

As explained in section 2.9.3, the advanced manual tuning procedure can be used to manually define the gain schedule table. You may define any number of gain scheduling values and have the Wizard interpolate the entire list. For dual loop tuning, you may use a list previously defined during velocity loop tuning.

To perform advanced manual tuning for the dual loop:

1. From the **Step 1: Select the Tuning Type** drop-down list of the Tuning Dual Loop dialog box, select **Advanced Manual Tuning**. The dialog box will be redisplayed as follows:



If you performed gain scheduling during velocity tuning (or at any other time), the previously-defined velocity KP and KI values will be displayed here, in the **Vel. Loop KP** and **Vel. Loop KI** columns.

> **Note:**
> To use a gain schedule table created through auto-tuning or from a previous advanced manual tuning session, click **Load GS Table** at the bottom and select the desired file from the Open dialog box.

2. The recommended procedure for determining the appropriate gain values for *a specific velocity* is as follows:

a. Be sure that the **Gain Scheduling ON** check box (in the middle) is not selected and that the Advanced Filter Designer is **OFF**.

b. Select an existing velocity or enter a new velocity value in a row appropriate to its place in the velocity range.

c. Enter the relevant gain coefficients for that velocity.

d. In the **Step 3: Set Test Parameters** and **Step 4: Set Record Parameters** blocks, be sure that the required values are entered.

e. Click **Run Test**. The system will send the gain coefficients for the selected row to the drive and the motor will operate at the selected velocity and according to the test and record parameters that you selected. The recorder will automatically record the feedback and display the results for the highlighted row.

f. Evaluate the results and, if needed, modify your values, repeating steps b. through e. until you are satisfied that the gain values for that velocity are acceptable. You may then highlight the row to indicate that it has been tested satisfactorily. To do so, click the right mouse button and from the shortcut menu, select **Toggle Bookmark**. The row with the bookmark will be highlighted in a light color.

g. Repeat steps b. through f. for one or more other velocity values until you are satisfied that you have determined a representative amount of recommended gain scheduling values.

3. You may have the Wizard complete the gain schedule table for you and then *test the entire table with gain scheduling*. To do so, perform the following procedure:

a. Check the **Gain Scheduling ON** check box.

b. In the row indicating the lowest velocity value for the range to be filled in, and in the row indicating the highest value in the range, select **Accept**.

c. To have the Wizard "complete" the gain scheduling for that range, click the right mouse button and from the shortcut menu, click **Interpolate Pos. Loop**.

The Wizard will calculate the coefficient values in the range you defined in step b. Rows before and after that range will take the values of the first and last **Accept**ed row, respectively.

d. To test the interpolated values, select a row of the table that indicates the velocity to be reached during the test. Alternatively, in the **Speed** text box, enter the velocity value. If that exact value does not appear in the table, the value closest to it — but not exceeding it — will be selected in the table.

e. In the **Step 3: Set Test Parameters** and **Step 4: Set Record Parameters** blocks, be sure that the required values are entered.

f. Click **Run Test.** The system will send the gain coefficients to the drive and activate the gain scheduling. During the test, the motor will turn according to the test and recording parameters. The recorder will be activated immediately in order to record and display the results of the position, position command and current command.

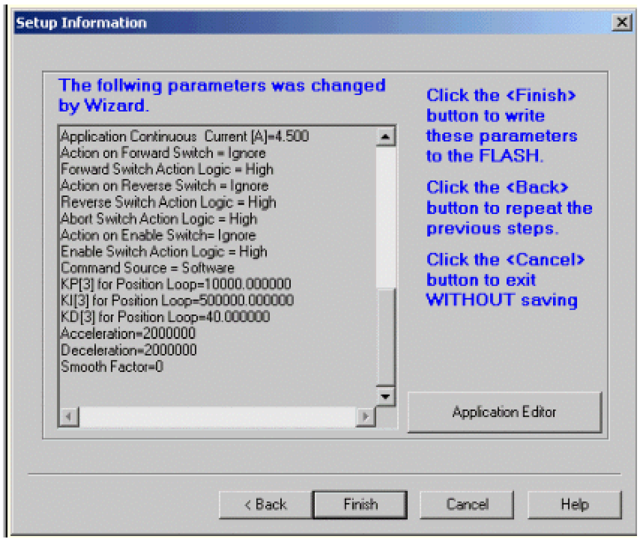
	Remember:
	▪ You can remove bookmarks using the **Clear All Bookmarks** option from the right-click shortcut menu. ▪ You can also automatically accept all rows of the table by selecting the **Accept ALL** option from the shortcut menu. (Use **Clear All Accepts** to remove all checks.)

4. To modify the gain schedule table further, repeat steps 2 and 3 as necessary.

5. To save your final gain schedule table, click **Save GS Table**. In the Save As dialog box, enter a name for the file, browse to the location at which it should be saved, and click **Save**. The system will save the table, along with its bookmarks.

2.12 Saving Your Application

Upon completion of your last tuning step, the Setup Information dialog box will be displayed, as in the following example:



This final dialog box lists all the parameters that were modified during the Wizard auto-tuning session. You may scroll through the parameters in the list at the left or you can open the Application Editor (explained fully in section 3.6) to view them as needed.

To finalize your new or modified application:

1. Click **Application Editor** if you wish to view the full set of application parameters in the Composer Application Editor. A standard Save As dialog box will be displayed for entering the path and file name for saving the file. The Application Editor will then be opened in read-only mode.

2. To change any of the application parameters, click **Back** to return to the previous Wizard process. Continue to click **Back** until you reach the tuning stage in which the parameter(s) need to be changed.

	Note: Clicking **Cancel** at any stage of this final process will restore all the parameters that were modified in this Wizard session and will complete the Wizard process.

3. When you are satisfied that all the setup information is final, click **Finish**. The Save As dialog box will be displayed for you to enter the path and file name for saving the file. The actual data will be saved in the drive memory. This step may take some time, and the progress will be indicated by a status bar displayed by the Wizard. Upon completion, the Composer Smart Terminal (section 3.3) will be displayed.

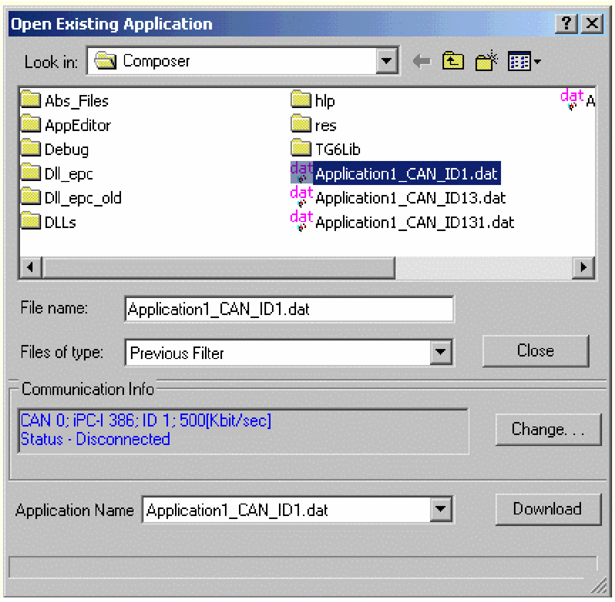
2.13 Composer Shortcuts

2.13.1 Opening an Existing Application

To access a previously-saved application:

- By clicking **Open an Existing Application** from the Welcome to Composer Application window displayed when you first access the Composer (section 2.2)
- By clicking the button in the Composer toolbar
- By selecting **File – Open New Application** from the Composer menu bar

The Open Existing Application dialog box will be displayed:

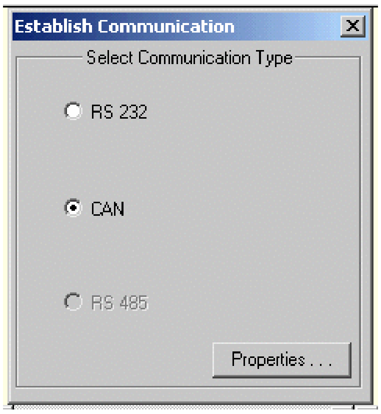


This dialog box can list two types of database files:

- Access-type files with .mdb file formats. These types of files may contain multiple applications in a single file.
- Binary file .dat files. These files contain a single application in each file, including a full set of the controller parameters (as saved in the flash memory), host (PC) communication parameters and the user program, if one exists.

To open an existing application:

1. Use the **File Name** text box and the **Files of Type** drop-down list to browse to the application file that you wish to open. The **Communication Info** block will display the host communication parameters that were stored for the selected database file.
2. To change the communication parameters for the selected application, click **Change** and select the communication type from the Establish Communication dialog box.



You may select your communication type and further define the exact parameters by clicking **Properties**, which displays the RS-232 Properties dialog box (section 2.3.1) or CANopen Properties dialog box (section 2.3.2) respectively.

3. Click **Download** at the bottom right of the Open Existing Application dialog box to open the selected application. The **Download** button will change to **Break** to enable you to stop the process if you are working with RS-232 communication. The process cannot be stopped with CANopen networks.

> Note:
> If your drive supports a binary format download and you select an .mdb-type file, the Composer will offer to convert the application to a .dat format.

4. Edit your application as necessary. Save it according to the procedure explained in section 2.12.

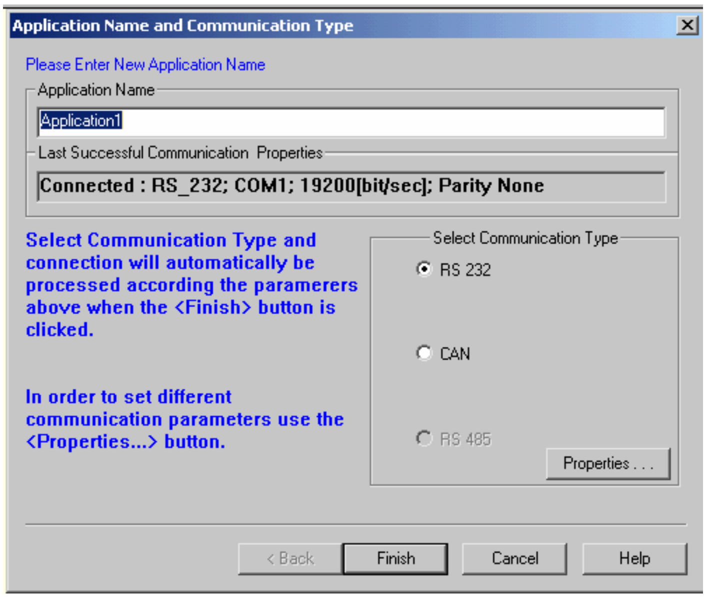
2.13.2 Opening Communication Directly

You may activate the communication network directly to either define a new application or to use an existing one.

You may do this in one of three ways:

- By clicking **Open Communication Directly** from the Welcome to Composer Application window displayed when you first access the Composer (section 2.2)
- By clicking the  button in the Composer toolbar
- By selecting **Communication – Open Communication Directly** from the Composer menu bar

The Application Name and Communication Type dialog box will be displayed:



To define your communication network:

1. In the **Application Name** textbox, type a name for the application or use the default name displayed.
2. In the **Select Communication Type** block, select the communication network to be opened. To further define the network parameters — or to communicate with more than one drive — click **Properties**, which displays the RS-232 Properties dialog box (section 2.3.1) or CANopen Properties dialog box (section 2.3.2) respectively.
3. Once you have selected your communication parameters, click **Finish**.

2.13.3 Loading the Network

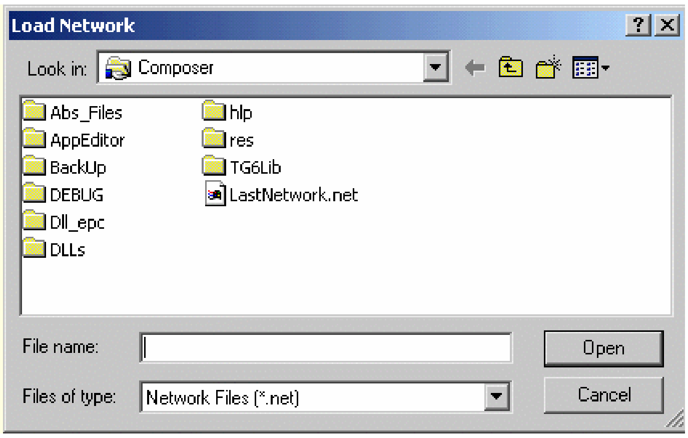
This option enables you to configure the Composer to communicate with several drives simultaneously regardless of communication type. To load a network, you need to first create the network .net file by establishing communication with the required drives.

To create a network file:

1. Select **Communication – Network – Save Network** from the Composer menu bar. The Save Network dialog box will be displayed.
2. Browse to the location at which the file should be saved, enter a name for the file and click **Save**.

To load a network:

1. Either click **Load Network** from the Welcome to Composer Application window displayed when you first access the Composer (section 2.2), or select **Tools – Network Load Network** from the Composer menu bar. The Load Network dialog box will be displayed:



2. From the **Files of type** drop-down list, select **Network Files (*.net)**.
3. Browse to the location of your network file.
4. Click **Open**. The Composer will load the selected network. If communication with one of the drives fails, a **Can't open communication** message will be displayed with the parameters of that drive. Click **Abort** to stop the network loading process, **Retry** to attempt to reload the network, or **Ignore** to continue establishing communications with the other drives in the network.

Chapter 3: Using the Composer

Once you have completed the initial setup of your Elmo servo drive(s), you can use the Composer for a wide range of on-going motion control processes. The Composer tools are accessed directly through the Composer toolbar buttons or via the menu bar options.

3.1 The Composer Desktop

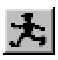
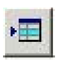
When you access the Composer application, the Composer Wizard Welcome dialog box enables you to directly open an existing application, communication or an existing network. To access other Composer tools, first close the Welcome dialog box.

3.1.1 The Toolbar

The Composer toolbar contains buttons that enable you to quickly access the most frequently-used tools and options in the Composer application.

*Figure 3-1: **The Elmo Composer Toolbar***

Table 3-1 lists each toolbar element and its function.

Button/List	Function
	Open a new application.
	Open an existing application.
	Save the open application.
	Select the user mode for the open application.
	Run the motor.
	Stop the motor.
	Access the Smart Terminal.
	Access the Motion Monitor.
	Access the Elmo Studio.
	Access the Table Editor.
	Access the Sync Manager.
	Access the Composer Wizard.

Button/List	Function
	Access the Scope (Graph Editor).
	Display Composer version information.
	Get context-sensitive help.
Application1_CAN_ID5	Select an active communication option (and application)
	Open communication directly.
	Disconnect: close the open application.

Table 3-1: ***Toolbar Elements***

3.1.2 The Menu Bar

The menu bar along the top of the Composer desktop provides access to the full range of tools and Composer options, as described in Table 3-2. The main menu options change according to the Composer tools that are accessed.

Menu	Option	Description
File	New Application	Create a new drive application.
	Open Application	Open a previously-created application.
	Save Application	Save the open application.
	Close	Close the open application.
	Exit	Close the Composer application.
View	Toolbar	Display / Hide the Composer toolbar.
	Status Bar	Display / Hide the status bar at the bottom of the desktop.
Communication	Open Communication	Open the communication network directly.
	Disconnect	Close an open application.
	Disconnect All	Close all open applications.
	Network	Save the open network / Load a previously-created network.
Tools	Smart Terminal	Display the Smart Terminal for a selected application.
	Motion Monitor	Display the Motion Monitor for viewing motion parameters in real time.
	Elmo Studio	Access the Elmo Studio for use with the open application.

Menu	Option	Description
Tools, cont.	Table Editor	Access the Table Editor for downloading a table into the driver. Select **Table Editor - Table Browser** to open a PVT or PT table file for editing. To create a new table in the Table Editor, select **Table Editor - PVT** or **Table Editor - PT**, as appropriate.
	Sync Management	Access the Sync Manager to perform synchronization through the CANopen network.
	Wizard	Display System Database dialog box for performing tuning steps with the Wizard.
	Advanced Manual Tuning	Display the relevant dialog box for performing advanced manual tuning directly from the Composer main menu.
	Firmware Download	Download the new firmware version to the driver.
	Scope	Access the Scope application.
Window	Cascade	Display open Composer windows in an overlapping fashion.
	Tile	Display open Composer windows one next to the other.
	Arrange Icons	Order the icons of open Composer windows.
Help	Help Topics	Access the Composer online help program.
	About Composer	Display Composer version information.

Table 3-2: ***Menu Bar Options***

3.1.3 Getting Help

The Composer application provides two kinds of help:

- Detailed online help, which you access by selecting **Help** from the menu bar
- Short, "what's this?" help, which you use by clicking the button in the toolbar and then pointing to the item for which you need assistance

3.2 The Motion Monitor

The Composer Motion Monitor controls the recording function of the drive and can display nearly any relevant value of the system. It includes a recording utility that enables you to record any four parameters (eight with the Harmonica) during a time period that you define. To display the Motion Monitor, click the button in the toolbar or select **Tools – Motion Monitor**.

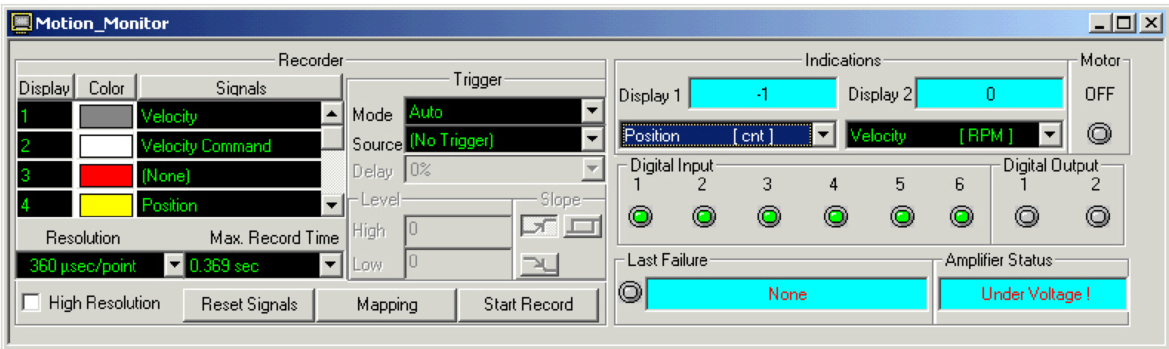


Figure 3-2: ***The Motion Monitor***

The Motion Monitor is divided into two main parts: the Recorder on the left and the Indications section on the right.

3.2.1 Motion Monitor Recorder

The Recorder is a monitoring and debugging tool for testing application and plant behavior. It uses the Scope application (section 3.5) to display the results of the tests in two graphs, as shown in the following example:

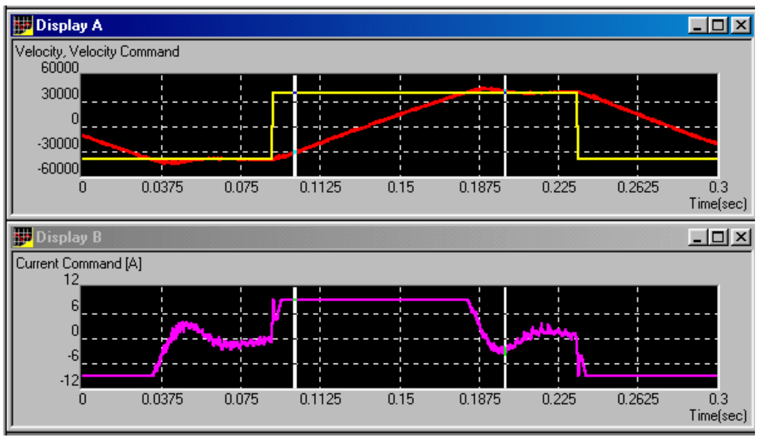


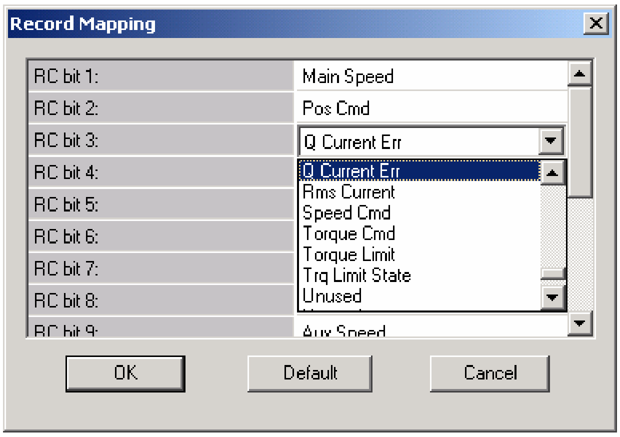
Figure 3-3: ***Vector 1 and Vector 2 Graphs***

To determine the graph parameters, you can select the signals to record for each graph (a total of eight selections). You do this from the rows of **Display**, **Color** and **Signals** drop-down lists, which provide the following signal options:

- None
- Active Current [A]
- Analog Input
- Auxiliary Position
- Auxiliary Velocity
- Current - Phase A
- Current - Phase B
- Current Command [A]
- DC Bus Voltage
- Digital Input
- Position
- Position Command
- Position Error
- Reactive Current [A]
- Velocity
- Velocity Command

To record your parameters:

1. Use the **Display**, **Color and Signals** drop-down lists to define the signals to be recorded. You may select up to eight signals, using the scroll bar to display each drop-down list. To revise your list from the beginning, click **Reset Signals** to reset all signals to **(None)**.
   To map the signals (vectors) displayed in the **Signals** drop-down list, click the **Mapping** button at the bottom and select the signals from the Record Mapping dialog box, as follows:



   From each **RC bit** row, use the drop-down list to select a signal to be displayed. You may press **Default** to display the Harmonica default list of signals. Up to 16 signals can be shown in the lists. When you have completed your selection, click **OK**. The selected list of signals will be displayed, in alphabetical order, in each **Signals** drop-down list.

2. From the **Resolution** drop-down list, select the recording resolution, which is defined by the sampling time of the controller. You may wish to change this value according to your current work mode: current or velocity.
3. From the **Max. Record Time** drop-down list, select the maximum recording interval; this is dependent upon the **Resolution** value.

4. In the **Trigger** block, select the trigger parameters — if any — that will initiate the recording:
   - **Mode**: type of trigger.
     Single enables all trigger parameters and **Auto** sets all trigger parameters to default.
   - **Source:** defines the event that will cause the recording to begin.
     If an analog source is selected, all trigger types and their levels are displayed. If a digital source is selected, the trigger type will be displayed as **On Window** and trigger levels will be disabled. If **No trigger** is selected, the entire **Trigger** block will be disabled.
     - Active Current in amperes
     - Analog Input 1
     - Analog Input 2
     - Current Phase A in amperes
     - Current Phase B in amperes
     - Current Command in amperes
     - DC Bus voltage
     - Position
     - Position Command
     - Reactive Current
     - Velocity
     - Velocity Command
     - Abort
     - Digital Input 1
     - Digital Input 2
     - Digital Output 1
     - Digital Output 2
     - FLS
     - Enable
     - RLS
     - Begin
     - Digital input combination

     If you select **Begin**, the trigger **Source** and **Level** options will be disabled; the trigger will be recorded when the BG command is sent.
   - **Delay**: trigger time delay for recording
     From **0**% to **100**%, of the **Max. Record Time**.
5. In the **Level** text boxes, enter the **High** and **Low** trigger levels.
6. Click the appropriate button to indicate the trigger **Type**:

Positive slope: Set the trigger and select **High** level. The trigger will be recorded when the source signal crosses the chosen level from low to high.

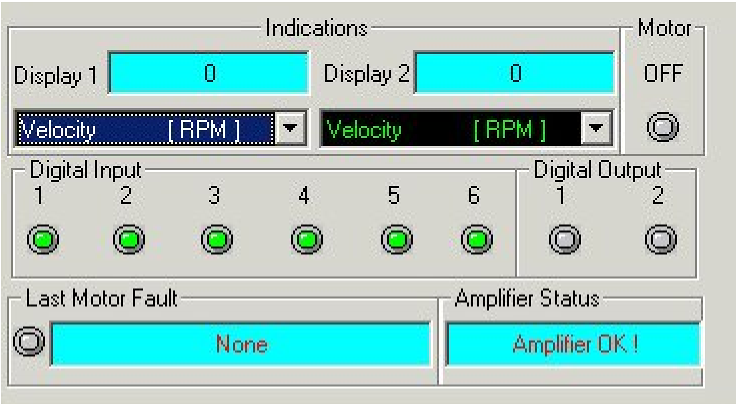
Negative slope: Set the trigger and select **Low** level. The trigger will be recorded when the source signal crosses the chosen level from high to low.

Window: Set the trigger and enable **High** and **Low** levels. The trigger will be recorded when the source signal crosses the chosen levels, as follows:
- The signal crosses the Low level twice.
- The signal crosses the High level twice.
- The signal crosses the Low level once and then crosses the High level.
- The signal crosses the High level once and then crosses the Low level.

7. To increase the recording resolution (to four times the sampling time of the standard recording resolution), you may click the **High Resolution** text box at the bottom left.
8. Click **Start Record** to begin the recording session.

3.2.2 Motion Monitor Indicators

This section enables you to enter parameters for displaying the current status of the drive. You use the two drop-down list boxes below the **Display 1** and **Display 2** text boxes to select two vectors for display.



For *linear* motors, your vector options are as follows:

- Active Current in amperes RMS
- Auxiliary Position in counts
- Position in counts
- Position Error in counts
- Velocity in counts/sec
- Velocity in m/sec
- Velocity Error in counts/sec
- Velocity Error in m/sec

For *rotating* motors, your vector options are as follows:

- Active Current in amperes RMS
- Auxiliary Position in counts
- Position in counts
- Position Error in counts
- Velocity in counts/sec
- Velocity in RPM
- Velocity Error in counts/sec
- Velocity Error in RPM

The current drive status is indicated by the row of LEDs.

3.3 The Smart Terminal

The Smart Terminal is an interactive mechanism that provides online communication with the servo drive. It enables you to send commands to a single axis and view the response. To display the Smart Terminal, click the button in the toolbar or select **Tools – Smart Terminal**. Once the connection with the application and communication has been made, the terminal will be displayed as follows:

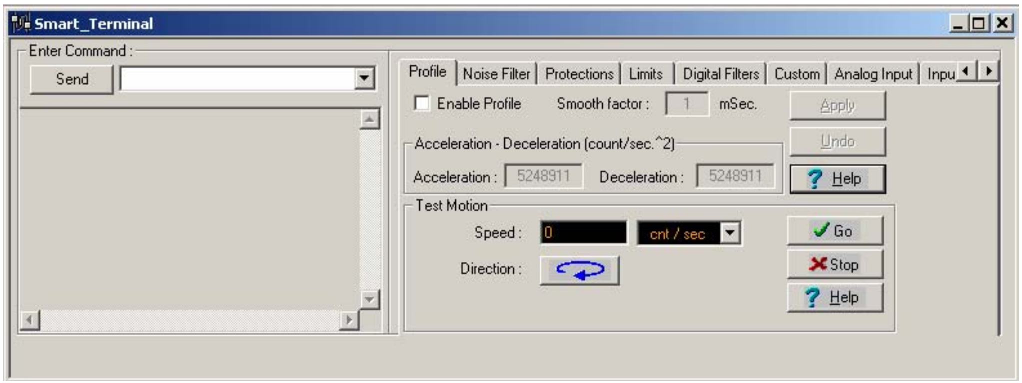


Figure 3-4: ***The Smart Terminal***

The Smart Terminal is divided into two main parts:

- The terminal on the left, which enables you to send commands to the drive, and displays messages from it
- The tabbed dialog boxes on the right, which you use to define the control commands

> Note:
> Refer to the Harmonica *Command Reference Manual* for detailed descriptions of the specific commands mentioned in this section.

3.3.1 Terminal

The terminal part of the Smart Terminal gives you direct interactive communication with the servo drive. You may key in commands in the text box, or select previous commands from the drop-down list, and then press **Send** to send the command. You can also copy a command from the command list and paste it into the text box for sending a second time.

3.3.2 Tabbed Dialog Boxes

Each dialog box is used to define different parameters for the Smart Terminal, sending the commands directly to the drive. Click the tab to display the dialog box.

> Notes:
> 1. When entering any value in any of the text boxes, be sure to click **Apply** in order to send the parameter to the controller.
> 2. Use the Mode drop-down list in the toolbar to switch between modes.
> 3. Values that you enter in the tabbed dialog boxes are not saved in the flash (or non-volatile) memory of the drive. In order to actually save the parameters, send the SV command from the Smart Terminal.

3.3.2.1 Profile Dialog Box

In velocity, position and dual loop modes, this dialog box is used to define the acceleration (AC command), deceleration (DC command), and smooth factor (SF command), as needed. It also enables you to test different motion parameters in current, velocity and position modes, using the **Test Motion** block at the bottom.

a. **Current Mode**
In current mode, the **Test Motion** block is activated in order to test the Torque (TC) command. When you click **Go**, the drive runs at the given torque (in amperes). Clicking **Stop** resets the torque value to default (0).

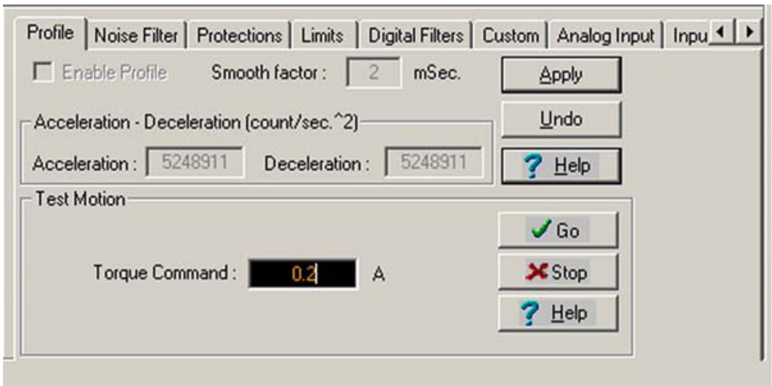


Figure 3-5: ***Profile Dialog Box - Current Mode***

b. **Velocity Mode**
In velocity mode, you may activate profiler (PM) mode by selecting **Enable Profile**.

Figure 3-6: ***Profile Dialog Box - Velocity Mode***

In the **Test Motion** block, you can define and test:

- **Speed** (JV command), in counts/second or RPM
- Motor **Direction**

When you click **Go** to start the test, the values you enter are sent to the drive and displayed in the command list at the left. If needed, the motor is first started (MO=1), and then the command is sent, followed by a Begin Motion (BG) command. Clicking **Stop** sends a Stop Motor command (ST) but does not actually disable the motor (MO=0).

	Note: To disable the motor during the test, click the STOP button.

c. **Position and Dual Loop Modes**

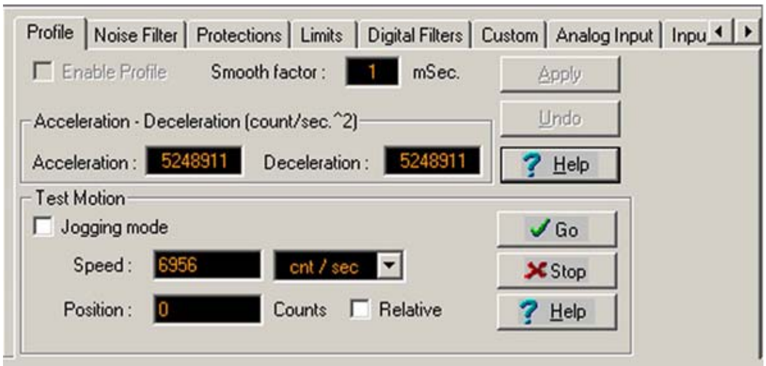


*Figure 3-7: **Profile Dialog Box - Position Mode***

In the **Test Motion** block, you can define and test:

- **Jogging mode** (JV command)
- **Speed** (SP command, for PTP mode)
- **Position - Relative** (PR) or **Position** - absolute (PA)

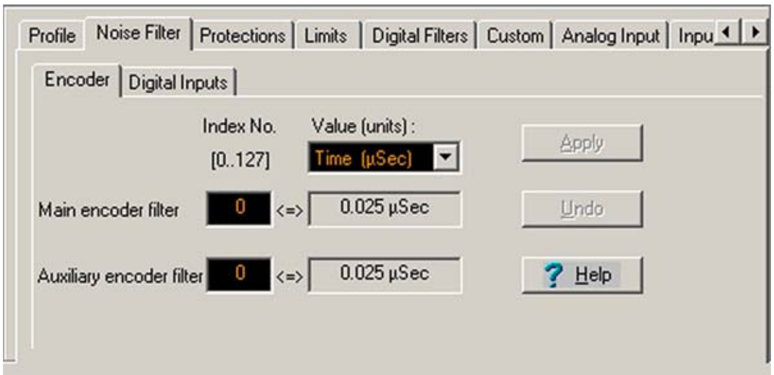
In these modes, the JV and PA commands can be sent only when the motor is on.

As in velocity mode, when you click **Go** to start the test, the values you enter are sent to the drive and displayed in the command list at the left. If needed, the motor is first started (MO=1), and then the command is sent, followed by a Begin Motion (BG) command. Clicking **Stop** sends a Stop Motor command (ST) but does not actually disable the motor (MO=0).

	Note: To disable the motor during the test, click the STOP button.

3.3.2.2 Noise Filter Dialog Box

This dialog box is used to define the filters for the main and auxiliary encoders, and the digital input filters.



In the Encoder tab, you need to enter the index number that indicates the filter level for each encoder. To the right of each dialog box is the actual **Value** corresponding to the index, in either sampling **Frequency** (Hz) or **Time** (microseconds) units. When **Frequency** is selected, encoder pulses larger than these values will not be sensed. When **Time** is selected, pulses shorter than the selected value will not be sensed.

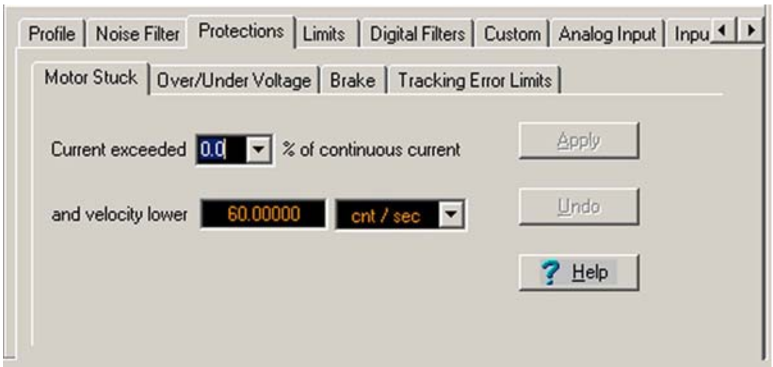
The software commands represented by each parameter are as follows:

- **Main encoder filter**: EF[1]
- **Auxiliary encoder filter**: EF[2]

In the Digital Inputs tab, for each digital input, enter the time period, in milliseconds, used to prevent switch bouncing. Input pulses of a duration than than IF[N] are rejected. Pulses longer than IF[N] are sensed. The six entries represent software commands IF[1] through IF[6].

3.3.2.3 Protections Dialog Box

This dialog box is used to define the range of protections available for the drive, as follows:



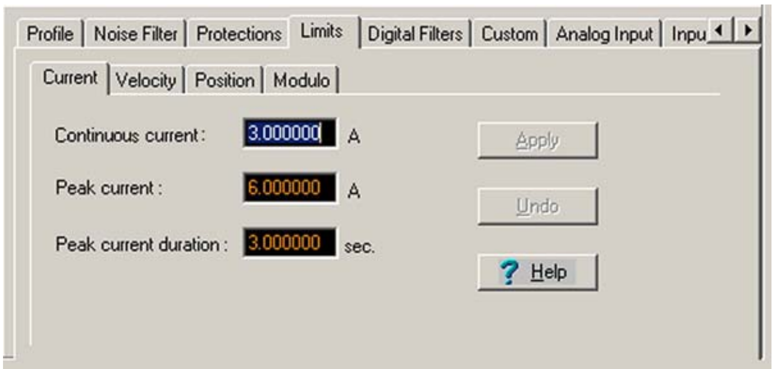
- Motor Stuck tab:
  - **Current exceeded x% of continuous current**: CL[2]
    Defines "motor stuck" as the tested torque level being a percentage of continuous current limit CL[1].
  - **and velocity lower**: CL[3]
    The absolute threshold main sensor speed under which the motor is considered not moving.
  - (When CL[2] is set to 0, the mode is deactivated.)

  For example, if the current is 50% of the continuous current (CL[2]=50) and the velocity is lower than (CL[3]=500) for 3 seconds, the drive will abort (MO=0).
- Over/Under Voltage tab:
  - **Maximum over voltage**: XP[1]
  - **Minimum under voltage**: XP[1]/8

  You may click the **Reset to default value** button to recall the default drive values.
- Brake tab:
  - **Brake stays released for x mSec after motor off**: BP[1]
    Defines the delay for engaging the brake after the motor is turned off (MO=0).
  - **Brake stays engaged after motor on**: BP[2]
    Defines the delay, in milliseconds, required to disengage the brake after the motor is turned on (MO=1).
- Tracking Error Limits tab:
  - **Velocity tracking error**: ER[2]
    Maximum allowed velocity error, in either counts/second or rpm.
  - **Position tracking error**: ER[3]
    Maximum allowed position error, in counts.

3.3.2.4 Limits Dialog Box

This dialog box is used to define the range of limits of the drive, as follows:

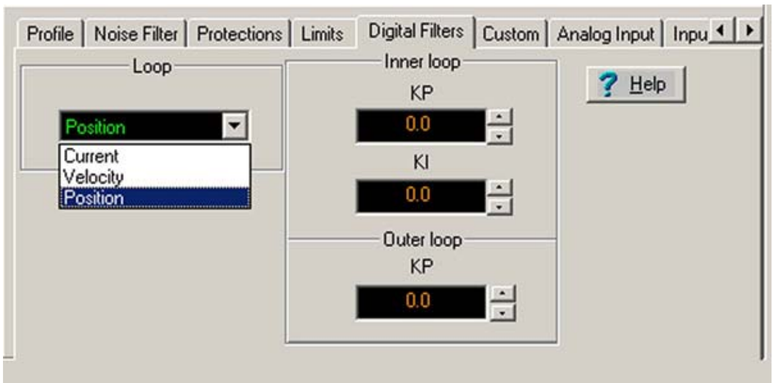


- Current tab:
  - **Continuous current**: CL[1]
    Maximum allowed continuous motor phase current.
  - **Peak current**: PL[1]
    Maximum peak current, in amperes.

  - **Peak current duration**: PL[2]
    Maximum peak duration, in seconds.
- Velocity tab:
  - **Command - Low**: VL[2]
    Minimum limit for speed command.
  - **Command - High**: VH[2]
    Maximum limit for speed command.
  - **Feedback - Low**: LL[2]
    Minimum limit of allowed motor speed.
  - **Feedback - High**: HL[2]
    Maximum limit of allowed motor speed.
  - **Stop deceleration**: SD
    Deceleration, in counts/second^2, used in the event of Stop Motor.
  - **Velocity unit**: **cnts/sec** or **RPM**
- Position tab:
  - **Command - Low**: VL[3]
    Minimum limit for position command.
  - **Command - High**: VH[3]
    Maximum limit for speed command.
  - **Feedback - Low**: LL[3]
    Minimum limit for allowed motor position range.
  - **Feedback - High**: HL[3]
    Maximum limit for allowed motor position range.
- Modulo tab:
  - **Main Feedback - Minimum**: XM[1]
    Minimum value of main-feedback counting range.
  - **Main Feedback - Maximum**: XM[2]
    Maximum value of main-feedback counting range.
  - **Auxiliary Feedback - Minimum**: YM[1]
    Minimum value of auxiliary-feedback counting range.
  - **Auxiliary Feedback - Maximum**: YM[2]
    Maximum value of auxiliary-feedback counting range.

3.3.2.5 Digital Filters Dialog Box

This dialog box is used to define the digital input filters for current, velocity and position modes. For each mode, you define the KP (proportional gain coefficient) and the KI (integral gain coefficient) for the PID filter.

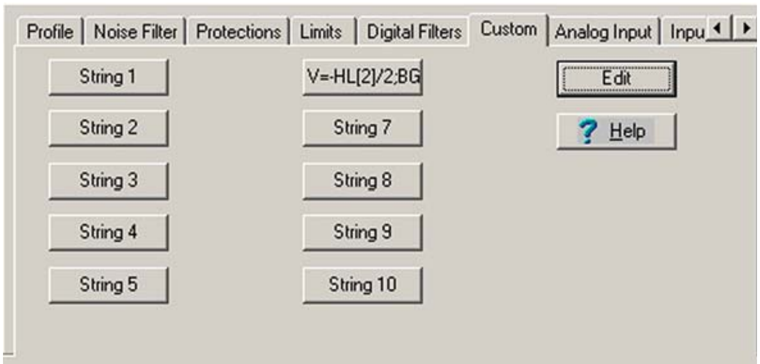


The options are as follows:

- Current mode: **KP** and **KI**
- Velocity : **KP** and **KI**
- Position: **KP** and **KI** for the inner loop and **KP** for the outer loop

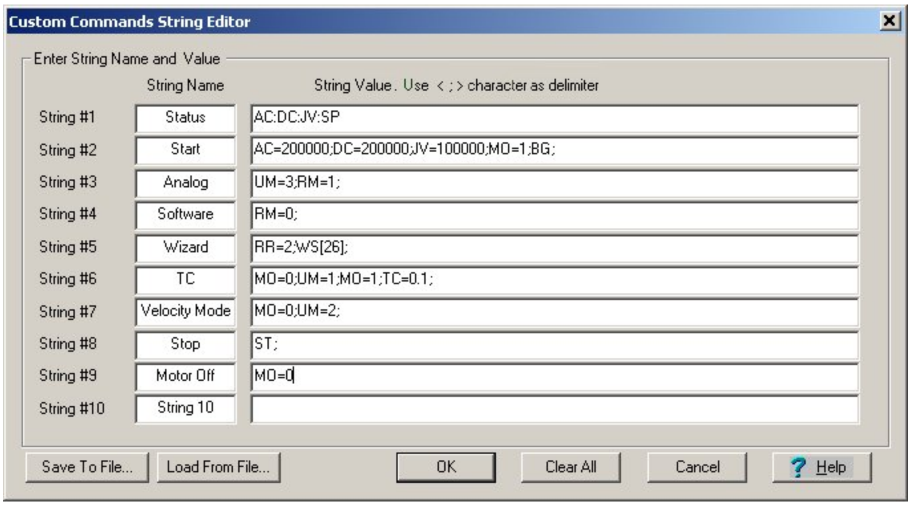
3.3.2.6 Custom Dialog Box

This dialog box enables you to program a set of command buttons to execute different commands or sets of commands. You may program up to ten command sequences, changing each button (command) name as needed.



To program a command button:

1. Click **Edit**. The Custom Commands String Editor dialog box will be displayed:

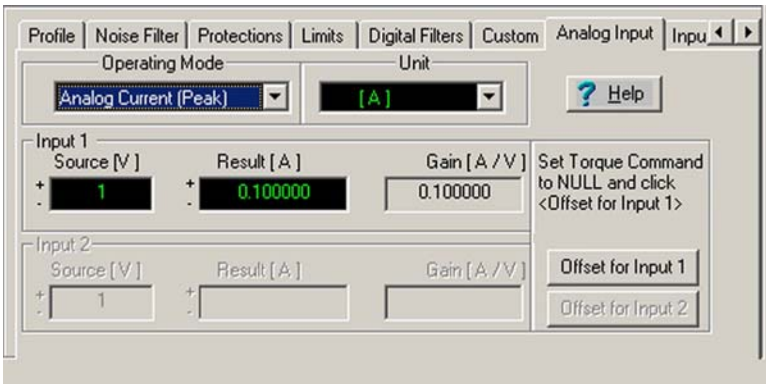


2. Select the **String #** (command) that you wish to overwrite and in the **String Name** text box, enter a name for the command (button).
3. In the adjacent **String Value** text box, type the command string. Be sure to separate multiple commands with a semicolon (;) and add a semicolon at the end of the string.
4. Repeat steps 2 and 3 to program additional commands.
5. When you complete your command definition, click **OK**.

You may save the set of commands as a text file by clicking **Save To File**, and subsequently recall the file by clicking **Load From File**.

3.3.2.7 Analog Input Dialog Box

This dialog box is used to set the gain values for the analog input, according to operating mode (velocity or current).



Fill in the dialog box as follows:

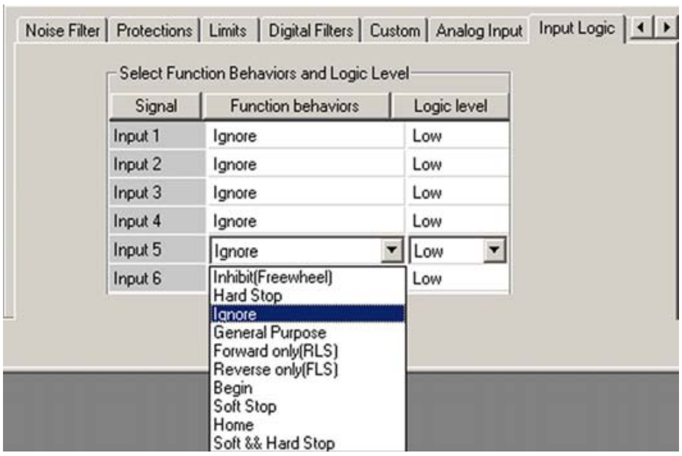
1. From the **Operating Mode** drop-down list, select the mode for working with the controller:
   - **Software**: Disables analog input.

- **Analog Current (Peak)**: Translates analog input (volts) to a current command (in amperes) in order to compute the gain.
- **Analog Velocity**: Translates analog input (volts) to a velocity command (in counts/second or rpm for rotary motors and counts/sec or meters/V for linear motors, selected from the **Unit** list) in order to compute the gain.

2. According to your selection in step 1, enter the required input values and press **<Enter>** on the keyboard.
3. In Analog Velocity mode, to determine the analog input offset, short the input to ground (Velocity command) and then click **Offset for Input 1** at the bottom right.

3.3.2.8 Input Logic Dialog Box

This dialog box is used to define the actions that should occur when the various mechanical limit switches are activated by an incoming signal.



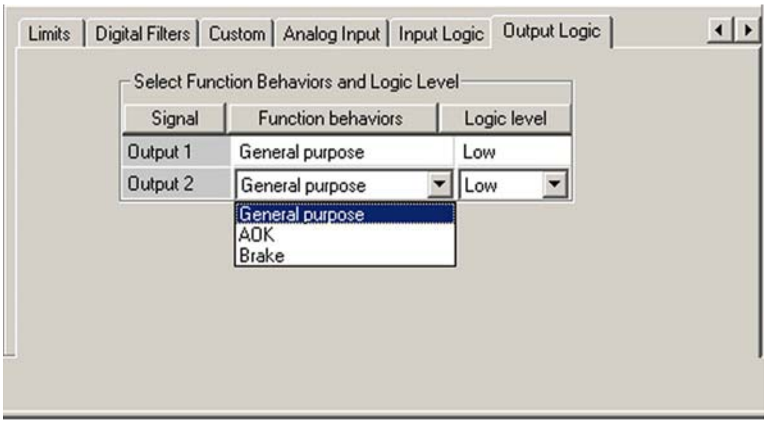
For each **Signal**, select the **Function behavior** and **Logic level** that will activate the response. Six signals are available, accessed by using the scroll bar. The options for the switches are as follows:

- **Function behaviors**:
  - Inhibit (Freewheel)
  - Hard Stop
  - Ignore
  - General Purpose
  - Forward Only (RLS)
  - Reverse Only (FLS)
  - Begin
  - Soft Stop
  - Hard & Soft Stop
  - Home (*available in Input 5*)
  - AUX Home (*available in Input 6*)
- **Logic level**: **High** or **Low**

For details about the various function behaviors, refer to Table 2-2 in section 2.5.

3.3.2.9 Output Logic Dialog Box

This dialog box is used to define the actions that should occur when the various mechanical limit switches are activated by a digital output signal.



For each **Signal**, select the **Function behavior** and **Logic level** that will activate the response. Two signals are available, with the following options:

- **Function behaviors**:
  - **General Purpose**
  - **AOK**
  - **Brake**
- **Logic level**: **High** or **Low**

For details about the various function behaviors, refer to Table 2-3 in section 2.5.

3.4 The Elmo Studio

The Elmo Studio is a basic application that provides program creation and editing options for software programs, including Upload, Download, Compile and Execute Program. You can use it in conjunction with the Composer to edit the application programs. The Elmo Studio application is fully described in Chapter 4 of this manual.

3.5 The Scope

The scope is a graphic display tool that enables you to view the data that has been recorded by the Motion Monitor. It displays multiple recorded vectors in the same window or in separate windows, and can generate new data vectors by applying arithmetic operations on the existing data vectors. You may use the scope to view and analyze recorded motions, zooming in and out of the graphs. You may add required text to the graphs, retrieve statistical information (such as average and maximum values) and calculate step response parameters (such as bandwidth and damping).

The Scope window is displayed automatically when the Composer displays graphs of data recorded in the various Wizard dialog boxes or in the Motion Monitor. Two graphs are displayed in the Scope data viewer, as in the following example:

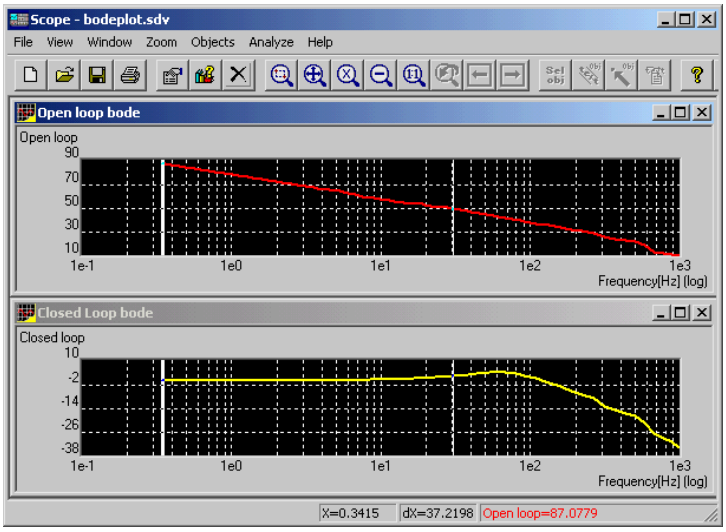


Figure 3-8: ***The Scope Window***

3.5.1 The Scope Toolbar

The Scope toolbar provides direct access to the main menu functions, enabling you to change your zoom options and to manipulate the displayed data graphs.

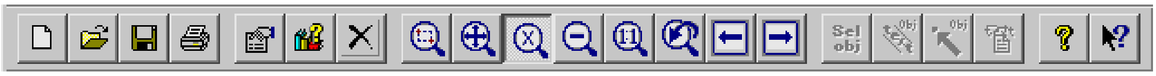
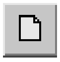
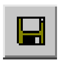
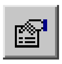
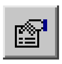
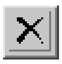
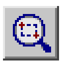
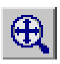
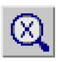


Figure 3-9: ***The Scope Toolbar***

Table 3-3 describes the function of each button on the Scope toolbar.

Button	Function
	Open a new window.
	Open an existing graph.
	Save the displayed graph.
	Print the displayed graph.
	Display the window properties.
	Organize the open display.
	Close the selected window.
	Zoom to the markers.
	Zoom in, in both directions.
	Zoom in.

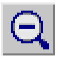
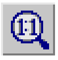
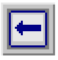
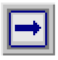
	Zoom out.
	Zoom to full size.
	Undo last zoom.
	Move left.
	Move right.
	Display information about Scope version.
	Display on-line help.

Table 3-3: ***Scope Toolbar Buttons***

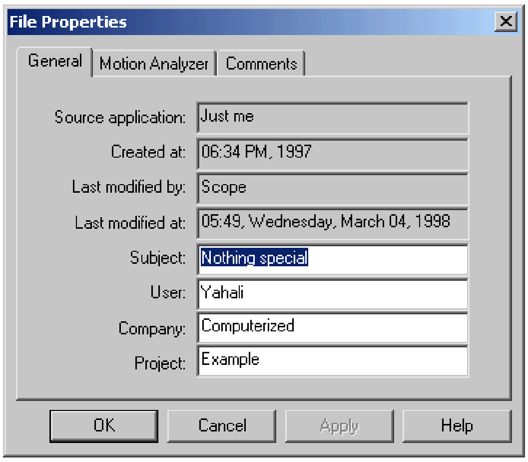
3.5.2 Using the Scope Menu

You can perform a wide range of operations through the Scope menu. The main scope functions are described in this section. Further operating instructions for the Scope are available using the **Help** menu in the Scope menu bar.

3.5.2.1 File Menu

The Scope **File** menu contains standard options for opening, saving and printing data files. Scope files are saved (and subsequently opened) as .sdv files. The **File** menu also includes:

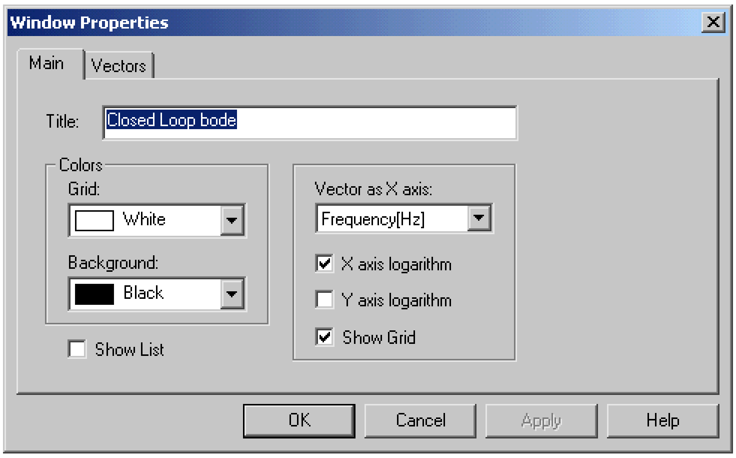
- **Import Data** and **Export Data**: for importing or exporting graph data to or from text (.txt) or Matlab (.mat) files.
- **Properties**: displays dialog boxes for entering additional information about the graph data, as in the following example:



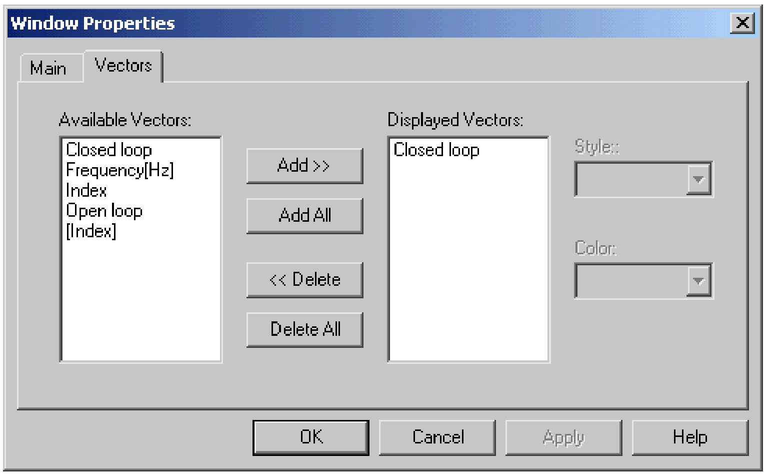
3.5.2.2 Window Menu

In addition to standard options — such as **Cascade**, **Tile** and **Arrange Icons** — the following options are included in this menu:

- **New** or **Properties**: displays the Window Properties dialog boxes for manipulating the graph grid.



Use this dialog box to name your graph (**Title**), select a color for the **Grid** and for the **Background**, and select the X-axis vector (**Vector as X axis**). Clicking the **Show List** option displays a list of the available vectors, which you define in the Vectors dialog box:



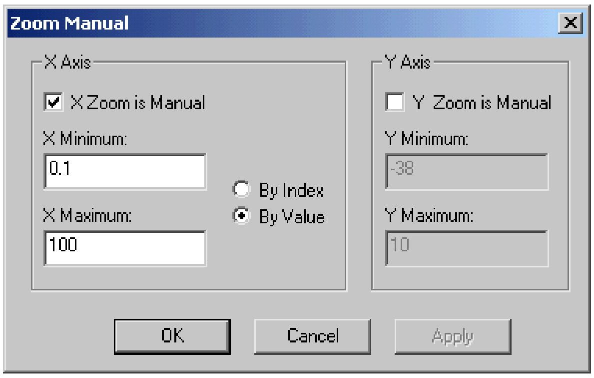
From the **Available Vectors** list, select the vectors that are to be displayed in the list, using the **Add** or **Add All** buttons. Then click **OK**. When you have created the list and entered all the Properties information, click **Apply** to implement the changes on the graph.

- **Markers**: displays grid lines at each marker on the graph. Dragging the marker lines displays their coordinates in the status bar at the bottom of the window.

3.5.2.3 Zoom Menu

Standard means of zooming into items on the graph include:

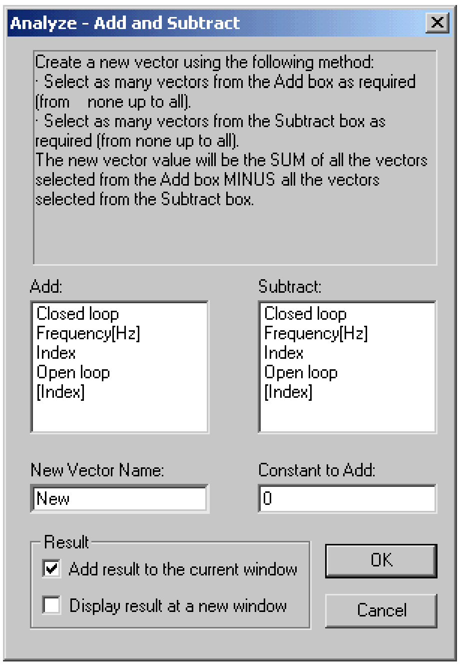
- Zoom **To Markers**: increases or decreases the zoom to the marked segment of the graph. Zoom **Out** is similar, in the direction of increasing the zoom.
- **Zoom Manual**: displays the following dialog box for explicitly defining the zoom parameters:



3.5.2.4 Analyze Menu

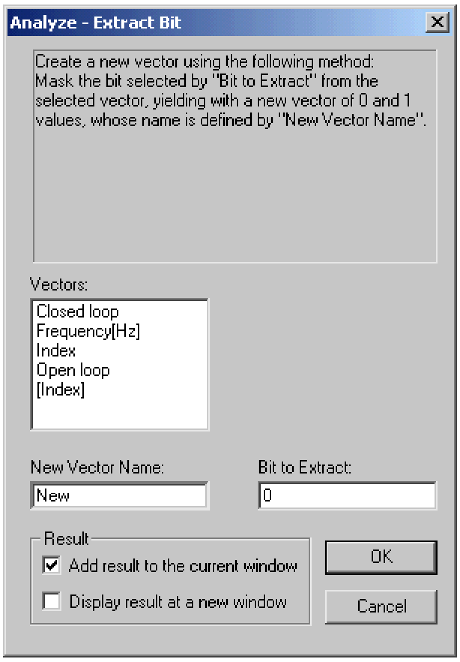
This menu is used to perform calculations on the graph data, as follows:

- **Add and Subtract**: calculates a new vector that is a sum of the vectors selected from the **Add** list, minus all the vectors selected from the **Subtract** list, then adding the value entered in the **Constant to Add** text box.

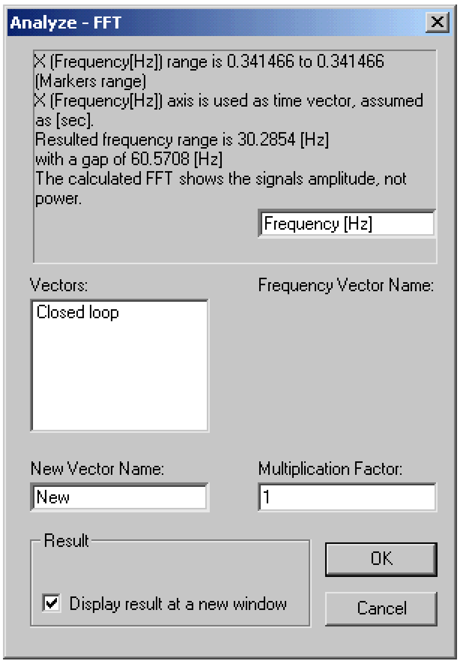


- **Multiply and Divide**: calculates a new vector by multiplying the vectors selected from the **Multiply** list and dividing by the vectors selected from the **Divide** List.
- **Over zero line**: defines a new vector by calculating the minimum value of the selected vectors, at each vector index.

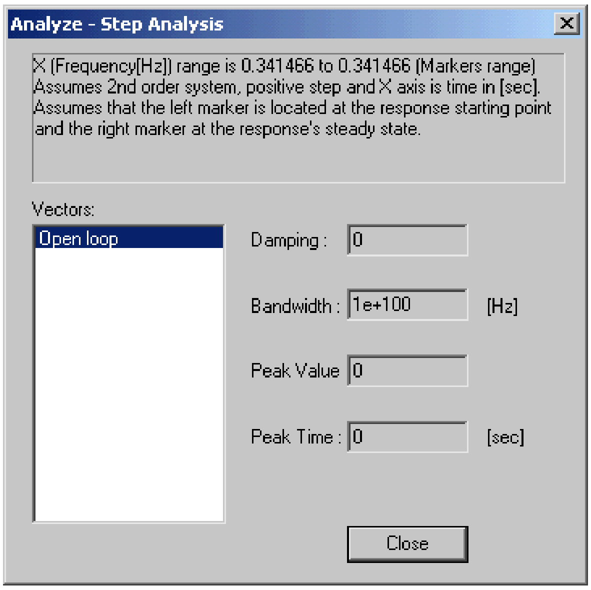
- **Under zero line**: defines a new vector by calculating the maximum value of the selected vectors, at each vector index.
- **Average**: defines a new vector by calculating the average value of the selected vectors at each vector index.
- **Extract Bit**: calculates the new vector by masking the bit defined in the **Bit to Extract** text box in the vector selected in the **Vectors** list.



- **Differentiate**: calculates a new vector by differentiating the vector selected from the **Source** list from the vector selected from the **Differentiate by** list. The **Multiplication Factor** is used to multiply the resulting new vector accordingly.
- **Integrate**: calculates a new vector by integrating the vector selected from the **Source** list into the vector selected from the **Integrated by** list. The **Multiplication Factor** is used to multiply the resulting new vector accordingly.
- **FFT**: calculates the FFT as in the following example:
  The X (time) range is 51.912 to 61.9416 (range of markers).
  The X (time) axis is used as the time vector, assumed to be in seconds.
  The resulting frequency range is 49.6032 Hz, with a gap of 0.193762 Hz.
  The calculated FFT displays signal amplitude, not power.



- **Statistics**: displays the properties of the selected value, including average **RMS** and **Tolerance** values (in percentage).
- **Step Analysis**: performs step analysis of the graph, assuming that this is a second-order system, with a positive step and x-axis in time in seconds. It also assumes that the left marker is located at the response starting point and the right marker at the response is steady state.



3.6 The Application Editor

The Application Editor enables you to view all the parameters in the application database and — in certain instances — to edit the data as well.

> Note:
> In order to maintain database integrity, it is highly recommended that you modify all application parameters through the Composer Wizard rather than using the Application Editor.

The Application Editor is displayed when you click **Application Editor** in the Setup Information dialog box which is accessible through the Composer Wizard (section 2.11). Below is an example of the main window of the Application Editor:

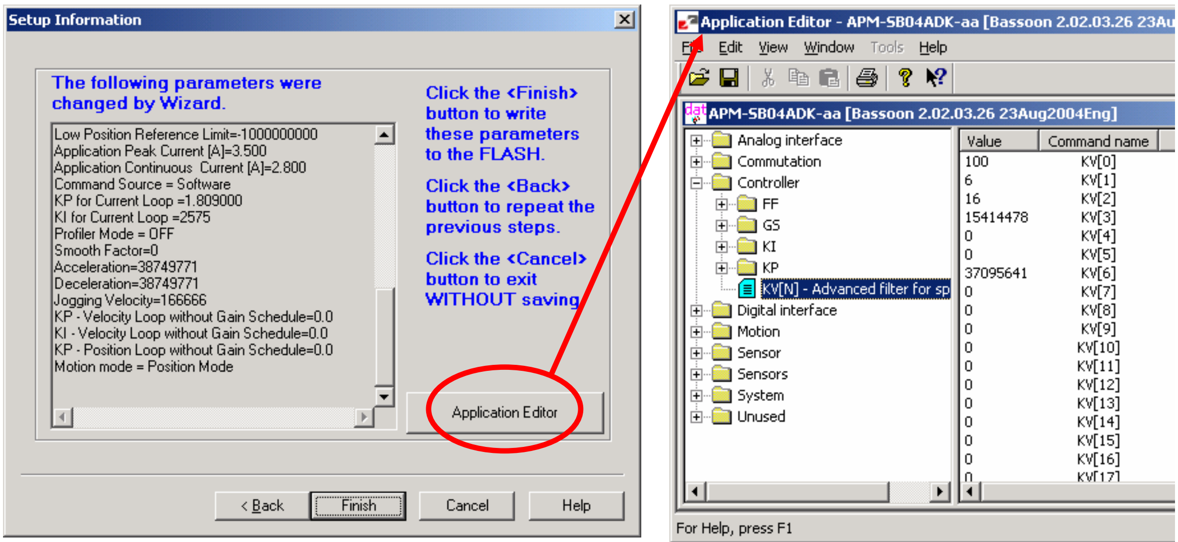


Figure 3-10: ***The Application Editor***

Always save your application before accessing the Application Editor through the Wizard.

The application parameters are arranged in groups according to definition given in the Command Reference file. The tree structure is displayed in the left window pane. Each folder represents a different definition group, with its parameters. All the properties of the selected parameter are displayed in the right pane. You can use the **Edit – Find** option to search for a specific parameter.

3.7 The Table Editor

You can use the Composer to download a PVT (Position, Velocity, Time) or PT (Position, Time) table to an attached servo drive, via CANopen communication. The Table Editor enables you to open an existing table that was created in an external spreadsheet program, make the required modifications to it, and then download it to the drive by selecting its node ID in the CANopen network.

> Note:
> This option is active only when CAN communication has been established.

3.7.1 Creating a PVT or PT Data File

You create a PVT or PT data file in Microsoft Excel and then open it in the Table Editor.

To create a data file in Excel:

1. Open a new Excel spreadsheet, using only a single sheet in your workbook. Prepare the row and line headings as in the following figures (PVT on left and PT on right):

	A	B	C	D
1	ID	Position	Velocity	Time
2	1			
3	2			
4	3			
5	4			
6	5			
7	6			
8	7			
9	8			
10	9			
11	10			

	A	B
1	ID	Position
2	1	
3	2	
4	3	
5	4	
6	5	
7	6	
8	7	
9	8	
10	9	
11	10	

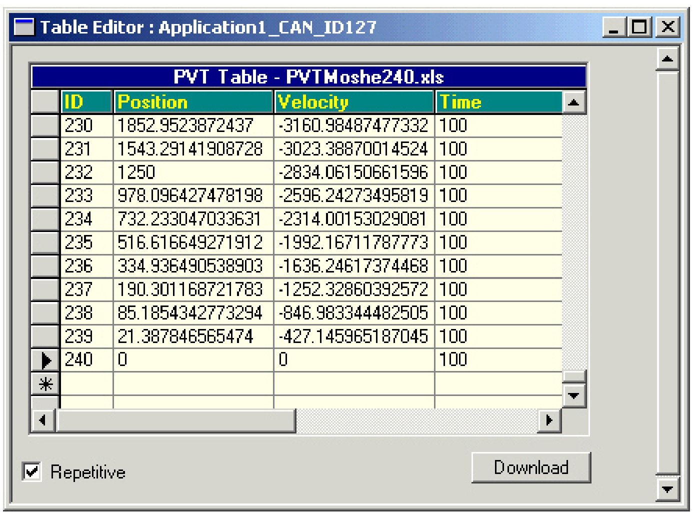
2. Save the file in the Composer directory.

3.7.2 Editing a File in the Table Editor

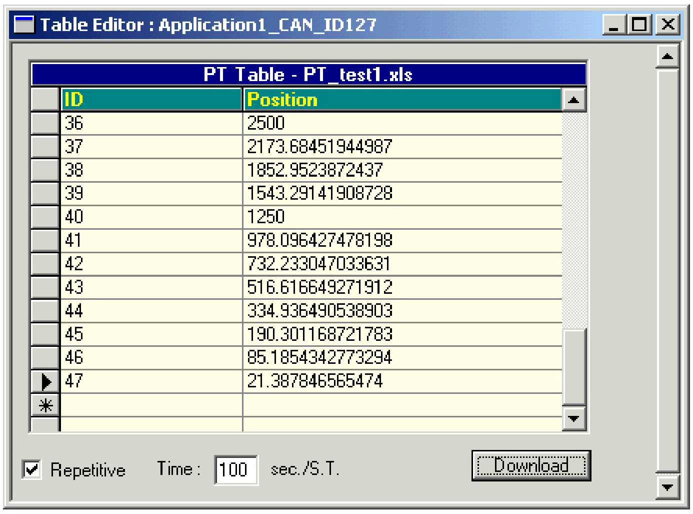
You may open an instance of the Table Editor for each drive connected to a node of the CANopen network.

To open the Table Editor (if not already open) for editing a file:

1. From the Active Communication drop-down list in the Composer toolbar, select the drive to which the table should be downloaded.
2. Click or select **Tools – Table Editor – Table Browser**. The Open dialog box will be displayed.
3. In the Open dialog box, browse to the Excel file, and click **Open**. The table file will be displayed as in the following examples:



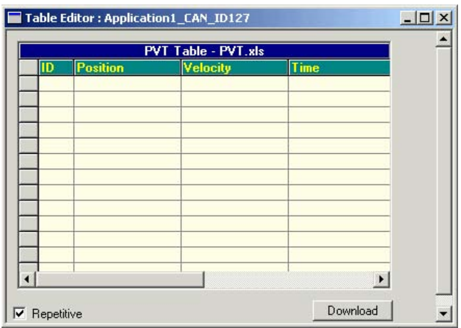
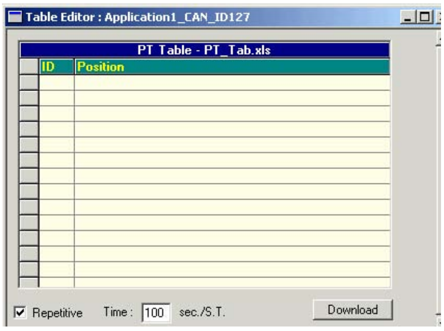
ID	Position	Velocity	Time
230	1852.9523872437	-3160.98487477332	100
231	1543.29141908728	-3023.38870014524	100
232	1250	-2834.06150661596	100
233	978.096427478198	-2596.24273495819	100
234	732.233047033631	-2314.00153029081	100
235	516.616649271912	-1992.16711787773	100
236	334.936490538903	-1636.24617374468	100
237	190.301168721783	-1252.32860392572	100
238	85.1854342773294	-846.983344482505	100
239	21.387846565474	-427.145965187045	100
240	0	0	100



The Table Editor dialog box contains the following elements:

- The PT or PVT table, with columns for node **ID**, and relevant parameters.
- **Repetitive** check box, to select/deselect repetitive mode.
- For PT tables, a **Time** text box to select the number of sample times (**S.T.**) per time unit.
- **Download** command button for downloading the table.

4. Edit the table as needed.

> Note:
> To open an "empty" PVT or PT file, you can select **Tools - Table Editor - PVT** or **Tools - Table Editor - PT**, as appropriate. A Table Editor window will be opened with the relevant empty table, as in the following examples:
>
> 
>
>
> 
>

3.7.3 Downloading a Table to a Drive

Once you have modified your PVT or PT table and saved it, you are ready to download it to the driver selected in the Active Communication drop-down list.

To download your table:

1. Select/deselect the **Repetitive** checkbox as needed.
2. In the Active Communication drop-down list, be sure that the CANopen node of the selected driver is correct.

3. Click **Download**. The motor will begin moving in PVT or PT motion mode according to the parameters in the downloaded table.

To download a table for other drives (connected to other nodes), open an instance of the Table Editor and table for each drive, selecting the drive node ID from the Active Communication drop-down list in the toolbar.

	Note: To download an open table to all nodes simultaneously, click the button in the toolbar. The table will be downloaded to all drives for which a Table Editor window was opened.

3.8 The Sync Manager

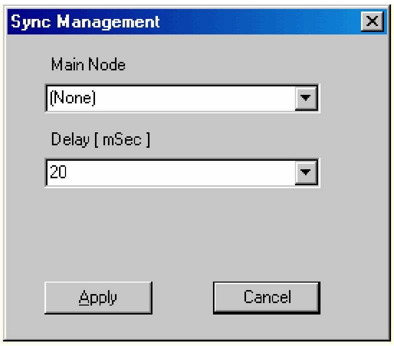
The Sync Management function operates within the Composer to synchronize the internal clock of all drives connected through the CANopen network. Only drives connected via CANopen can therefore be synchronized.

The Sync Manager operates as follows:

1. You select one of the CANopen nodes to be "Sync Master."
2. This node is used to transmit a synchronization message indicating the sync master's own internal 32-bit absolute time.
3. The Composer transmits this time stamp message to all nodes connected to the CANopen network, synchronizing the internal clock of all connected drives.

To perform synchronization:

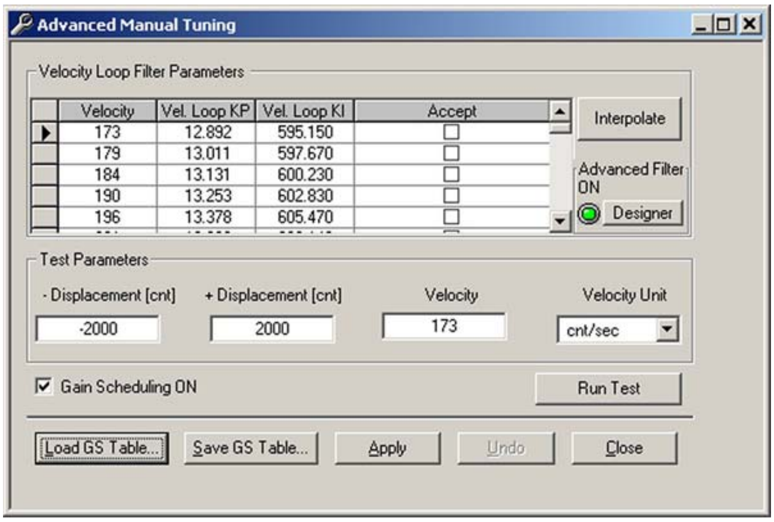
1. Activate the CANopen communication network by clicking the button in the toolbar or selecting **Communication – Open Communication** (section 2.12.2).
2. Click SYN or select **Tools – Sync Management**. The Sync Management dialog box will be displayed as follows:



3. From the **Main Node** drop-down list, select the node that is to serve as the Sync Master. This time stamp will be used to synchronize all other connected nodes.
4. From the **Delay** drop-down list, select the time interval after which each sync message should be sent.
5. Click **Apply**. The system will begin to send sync and time stamp messages.
6. To stop the message transfer, select **(None)** from the **Main Node** list and click **Apply**.

3.9 Advanced Manual Tuning

The **Tools - Advanced Manual Tuning** option provides direct access to performing advanced manual tuning of the velocity, position and dual loops. Using this option requires prior current and commutation tuning for the selected loop (shown in the drop-down Mode list in the toolbar). It enables you to run tests of the selected controller parameters with or without gain scheduling, as described in sections 2.8.3 (velocity loop), 2.9.3 (position loop) and 2.10.3 (dual loop). When you select this option, a dialog similar to the following (velocity tuning loop) is displayed:



In the top block, a gain scheduling table is displayed with functionality identical to that of the Wizard advanced manual tuning. This enables you to manipulate each of the 64 rows of entries, to test individual rows and to test the entire table using gain scheduling (click **Gain Scheduling OFF** to turn it on). You may use the **Load GS Table** and **Save GS Table** buttons to load a previously created table and to save the current table, respectively.

To run a test, be sure to first prepare the recorder and activate it prior to running a test. Clicking **Apply** sends all current values to the drive. (Clicking **Undo** reverts to the set of values sent at the last **Apply**.) Pressing **Run Test** turns the motor on and runs the test according to the displayed test parameters and selected gains.

If you have created a gain scheduling table and turned it off (**Gain Scheduling OFF**), when you click the **Close** button, the following message will be displayed to ensure that you do not cancel the use of the gain scheduling:

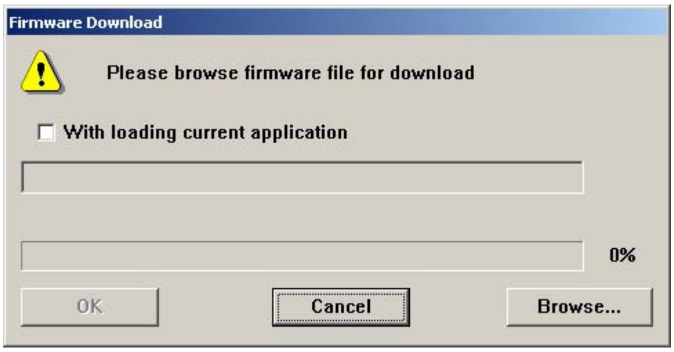
Gain scheduling is off. Manual filter parameters will be used. Continue?

To *use* gain scheduling, click **No**, toggle **Gain Scheduling OFF** back to **Gain Scheduling ON** and then click **Close** again.

3.10 Downloading Firmware

To download a new firmware version

1. Select **Tools – Firmware Download**. The Firmware Download dialog box will be displayed:



2. Type the full path and name of the data (*.abs) file to be downloaded, or click **Browse** and browse to the file. You may check the **With loading current application** option to indicate that the firmware should be loaded when the current application is downloaded. The loading process will begin and a status bar will indicate the progress. All other windows will be deactivated during downloading. Upon completion of the download, the following message will be displayed:

   Firmware Download was Completed Successfully!
   PLEASE REBOOT THE AMPLIFIER and CLICK OK!

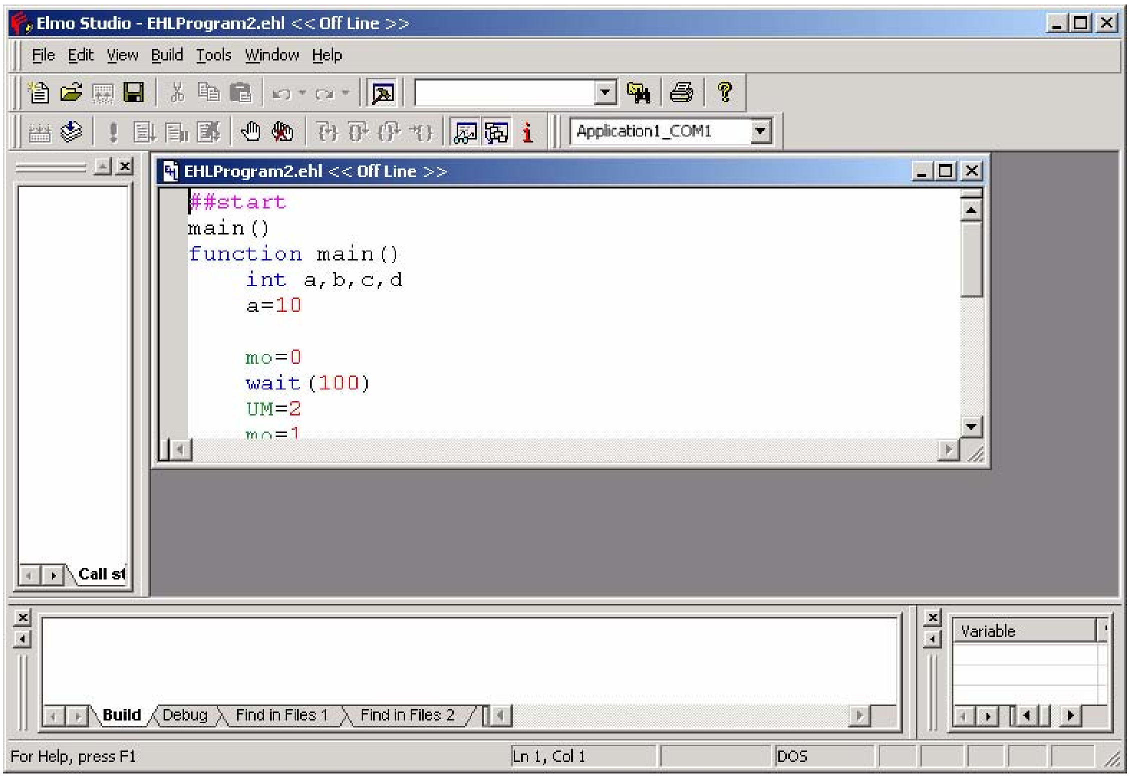
3. Reboot the drive and click **OK**.

To check that the firmware has been downloaded successfully, re-establish communication. Then use the Smart Terminal to send the VR command for checking the present firmware version number.

Chapter 4: Using the Elmo Studio

The Elmo Studio is a program editing application that includes a range of program creation, editing, build and debugging functions. You can use it in conjunction with the Composer to edit your application programs for subsequent download to the servo drive. The Elmo Studio is similar in look and general functionality to the Microsoft Visual Studio; users should already be familiar with the basics of source file coding, compiling and debugging.

To access the Elmo Studio, click the  button in the Composer toolbar or select **Tools – Program Editor.** The Elmo Studio desktop will be displayed as in the following example:



The last program to be edited will be displayed, and the name of the currently active application will be shown in the drop-down list box above it.

This chapter briefly describes the menus, toolbars and functionality of the Elmo Studio. You may wish to consult the relevant Elmo *Software Manual* for detailed information about specific program structure, definitions and limits for Elmo products.

4.1 The Elmo Studio Desktop

Upon accessing the Elmo Studio, you will see a set of open windows that can be opened and closed, and manipulated as needed. Across the top, as in most Windows applications, is the menu bar, with movable and customizable toolbars beneath it.

4.1.1 Desktop Windows

The following windows provide on-going information as you work in the Elmo Studio:

- Program window
  This is where the program being edited, compiled or debugged is displayed. In order to display multiple parts of this file at the same time, select **Window - New Window**. An additional pane will be displayed for you to view a different part of the same program.
- Stack window
  During debugging, all called functions that have not yet been returned are displayed here.
- Output window
  The Elmo Studio displays processing messages as follows:
  - Build tab
    Status messages from compiler and other tools during a build
  - Debug tab
    Messages from debugger to indicate run-time and other errors
  - Find in tabs
    Search results are displayed in the Find in 1 tab; subsequent searches can be displayed in the Find in 2 tab.
- Watch window
  This window provides a view of specified variables (that can be dragged-and-dropped from the Program window), along with their current values as they exist at the time the program is suspended.
- Communication Info window
  This pop-up window displays a list of the most recent system messages concerning communication between the host and the connected drive.

4.1.2 Elmo Studio Toolbars

The three toolbars — Standard, Build and Communication —contain buttons that enable you to quickly access the most frequently-used tools and options in the Elmo Studio application. You can move the toolbars around the desktop and relocate them for your convenience. You can also remove buttons and add others for commands that you frequently use (section 4.1.4.1).

Figure 4-1: ***The Elmo Studio Toolbars***

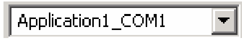
Tables 4-1 and 4-2 list each toolbar element and its function.

Button/List	Function
	Create a new program.
	Open an existing program.
	Upload a program from connected drive.
	Save the currently open program.
	Cut text from the program.
	Copy selected text in the program.
	Paste text into the program.
	Undo last action.
	Redo last "undo."
	Display/Hide (toggle) the Output window.
ST_56	Find displayed item.
	Find all occurrences of selected item.
	Print program.
	Get context-sensitive help.

Table 4-1: ***Standard Toolbar Elements***

Button/List	Function
	Build a program.
	Compile a program.
	Execute a program.
	Activate the debugger.
	Break.
	Kill the program.
	Insert/Remove (toggle) breakpoint.
	Remove all breakpoints.
	Step into.
	Step over.
	Step out.
	Run to cursor.
	Display/Hide (toggle) Watch window.
	Display/Hide (toggle) Stack window.
	Program limits.

Table 4-2: ***Build Toolbar Elements***

The Communication "toolbar" is actually the drop-down list — 

— that displays the names of every currently open application.

4.1.3 The Menu Bar

The menu bar along the top of the Elmo Studio desktop provides access to the full range of tools and options. The main menu options are described in Table 4-3.

Menu	Option	Sub-option	Description
File			Standard Windows options for opening, saving and manipulating program files, along with options to upload programs from and save programs to the connected drive.
Edit			Standard Windows Undo/Redo, Cut-and-Paste, and Find/Find All options.
View	Toolbars		Display/Hide Standard, Communication and Build toolbars.
	Windows		Display/Hide Output, Watch, Stack and Communication Info windows.
	Status Bar		Display/Hide status bar at the bottom.
	Active Line		Jump to and highlight Previous or Next active line of code.
Build	Compile		Compile a program source into an executable code.
	Build		Compile and download program.
	Kill Program		Stop program execution.
	Execute		Run program.
	Debug	Go	Run debugger.
		Break	Halt (suspend) program execution and return control to debugger.
		Set/Reset Breakpoint	Select/Cancel line for breakpoint.
		Clear Breakpoints	Delete all selected breakpoints.
		Step Into	Enter function and stop at first command.
		Step Over	Execute the next instruction line and then halt.

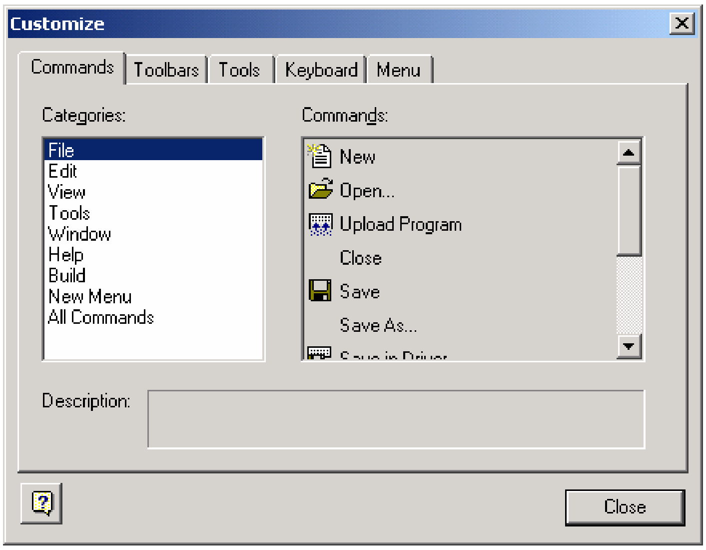
Menu	Option	Sub-option	Description
		Step Out	Complete the current function and then step out to the location immediately following the line on which the function was called.
		Run to Cursor	Halt execution at the instruction line at which the cursor is standing.
Tools	[user defined]		Range of applications that can be selected from the Customize - Tools dialog box.
	Customize		Display Customize dialog box for altering menu, toolbar and keyboard options.
	Options		Display Options dialog box to select Debug and Build parameters.
	Convert to New Format		Convert a program coded in Elmo .ell format to Elmo .ehl format.
Window	New Window		Open a new program window.
	Windows		Manipulate all open windows.
Help	Keyboard Map		Display a list of menu options, their accelerator key combinations and their descriptions.
	About Elmo Studio		Display information about the currently installed Elmo Studio version.
	Help Topics		Display Elmo Studio online Help.

Table 4-3: ***Menu Bar Options***

4.1.4 Customizing the Elmo Studio

The **Tools** menu contains two options for customizing your Elmo Studio application to your mode of work: the Customize dialog boxes and the Options dialog boxes.

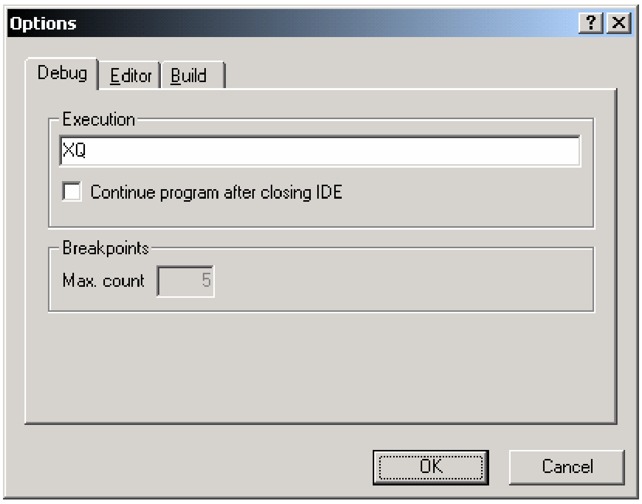
4.1.4.1 The Customize Dialog Boxes



The following tabbed dialog boxes are available:

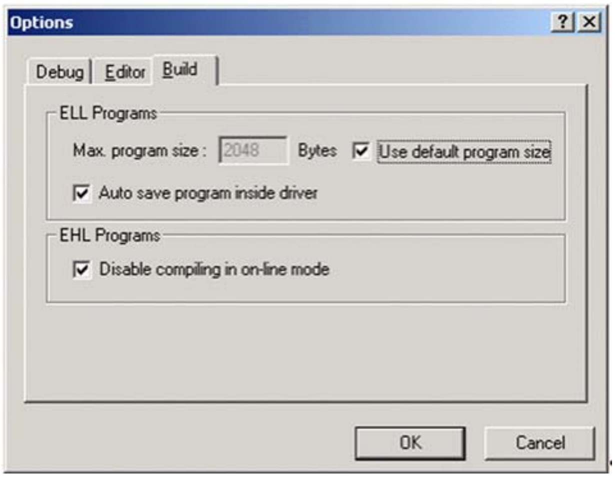
- Commands
  Enables you to add command buttons to the toolbar (by drag-and-drop) and to remove unneeded ones by dragging them off the toolbar.
- Toolbars
  For selecting which toolbars should be displayed or hidden, and to manipulate them as needed.
- Tools
  For adding and organizing frequently needed external applications to the Elmo Studio Tools menu.
- Keyboard
  For programming key combinations (“accelerators”) for frequently-used menu options.
- Menu
  For customizing the appearance of the various menu bars and popup menus.

4.1.4.2 The Options Dialog Boxes



The following tabbed dialog boxes are available:

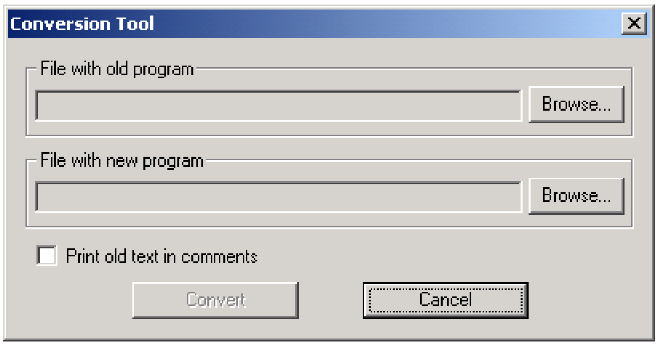
- Debug
  Enables you to select a command code for automatic debugging. Use the **Continue program after closing IDE** option to have the debug program continue to run even after the IDE has been closed.
- Editor
  For displaying/hiding the selection margin and the numbers margin. The selection margin is the gray column to the left of the program text, which enables you to select the entire line adjacent to the cursor click position. The numbers margin adds a column of line numbers to the left of the program text.
- Build



- For defining parameters of new and "old" programs. For .ell programs, the maximum program size is displayed, along with the option to use it as the default program size. Use the **Auto save program inside driver** option to automatically save the program in the drive memory. The **Disable compiling in on-line mode** option disables the independent **Build - Compile** menu option so that compiling is performed only as part of the Build function.

4.1.4.3 Convert to New Format

You can use the **Tools - Convert to New Format** option to save a program, coded in earlier Elmo .ell format, in Elmo .ehl format. The Conversion Tool dialog box is used to select the existing (.ell) file and save it under a new name, in the new .ehl format.



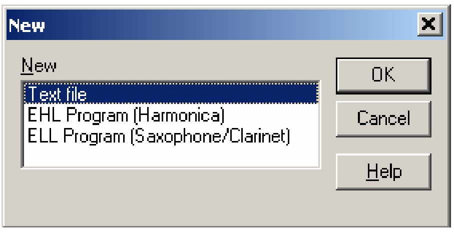
1. Click the **Browse** button next to the **File with old program** text box and navigate to the .ell file to be converted.
2. Click the **Browse** button next to the **File with new program** text box and navigate to the location at which the new file should be saved, giving it a new name if needed.
3. Click the **Print old text in comments** option if you wish to have the original text displayed, as comments, at the start of the new file.
4. Click **Convert** to activate the conversion process.

4.2 Elmo Studio Processes

This section describes the main functionalities of the Elmo Studio.

4.2.1 Creating a Program File

To write a new program for subsequent download to a drive, click  or select **File - New**. The New dialog box will be displayed:



1. Select the type of file to be opened:
   - **Text file** is a standard .txt file.
   - **EHL Program** is the file format used with Elmo Digital Servo Drives.
   - **ELL Program** is the older Elmo file format used with the Clarinet, Saxophone and Mini-Saxophone.
2. Click **OK**. A new program window will be opened for you to begin creating a new program.

4.2.2 Editing a Program File

To open an existing program file that resides on your computer, click or select **File - Open**. A program window will be opened with the selected file.

*To **upload** a program from a connected drive:*

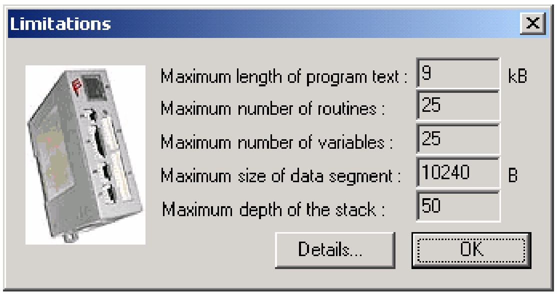
1. Click or select **Build - Upload Program**.
2. If no program file is currently open, the file will be uploaded and the Save As dialog box will be displayed for you to save the file on your computer. If a program file is currently open, the following message will be displayed: **Are you sure you want to upload the current file [file name]?** Click **Yes** to overwrite the existing file, or **No** to save the uploaded file under a different name.

Once the file is open, you can edit the program as you would in any text editor, using the tools available in the Standard toolbar. You can perform regular searches using the **Edit - Find** function, and you can search for multiple occurrences of an item using the **Edit - Find All** option (or clicking).

When using the **Find All** option, the results of the search will, by default, be displayed in the Find in Files 1 tab of the Output window. To perform subsequent searches without overwriting the results of a previous search, select the **Output to pane 2** check box in the Find All dialog box.

4.2.3 Defining Program Limitations

To view the various limitation values for your program, click the button to display the Limitations dialog box.



The dialog box is read only; click **Details** to display the full list of current limitation values.

4.2.4 Compiling a Program

Once your program is completed to your satisfaction, you can compile it. You may compile the program either online (while communication with the driver is active) or off-line. Alternatively, you may use the **Build** option to compile and download the program in one step (refer to section 4.2.5).

To compile a program, click or select **Build - Compile**. The Elmo Studio will compile the program and display all processing messages in the Build tab of the Output window.

When error messages occur during program execution and the program source needs to be fixed, you can double-click on the error message to locate the error in your program. When a Build operation finishes successfully with no errors, you can then execute the program (section 4.2.6) or run it using the Debug option (section 4.2.7).

4.2.5 Building a Program

The **Build** option enables you to have the Elmo Studio compile the program and automatically download it to the connected drive. To build a program, click or select **Build - Build**. The Elmo Studio will first compile the program. If no errors occur, it will then send the program to the Servo drive where it will be saved in the drive memory. If errors *do* occur during compiling, the Elmo Studio will record the errors and enter pointers in the program at locations where the errors occurred. It will then halt and will not download the program.

4.2.6 Running a Program

To run a program after it has been downloaded to a drive, click or select **Build - Execute**. The program will run independently of all debug options (such as Breakpoint and Step-by-step).

To stop a program while it is running, click or select **Build - Kill Program**.

4.2.7 Debugging

The Elmo Studio contains powerful tools for debugging the programs that you create and edit. It enables you to mark your program with breakpoints, and to control the debugging process according to your needs. Basically, you perform debugging according to the following steps:

1. Identify the section of the program where you suspect that a problem lies. This may be according to run-time error messages that you receive from the drive.
2. Mark the first instruction of that section with a breakpoint, clicking anywhere in the line and then clicking , pressing **<F9>** or selecting **Build - Debug - Set/Reset Breakpoint**. (You can cancel a breakpoint by repeating this action.)
3. You may also drag-and-drop variables from the program into the Watch window at the bottom. These are variables whose values you wish to know each time the program is suspended.
4. Start the debugging operation by clicking or selecting **Build - Debug - Go**. The debugger will execute the program until it reaches the first breakpoint, at which time program execution is halted. Each time the program you are debugging stops at a breakpoint, the debugger will update the Debug tag of the Output window with the relevant progress message. It will also indicate — with a yellow arrow and red highlight — the line of code at which the program stopped. Functions not yet returned will be displayed in the Stack window.
5. From here, you can use the relevant toolbar buttons or **Build - Debug** options (Table 4-4) to step through the program and continue the debugging operation manually.

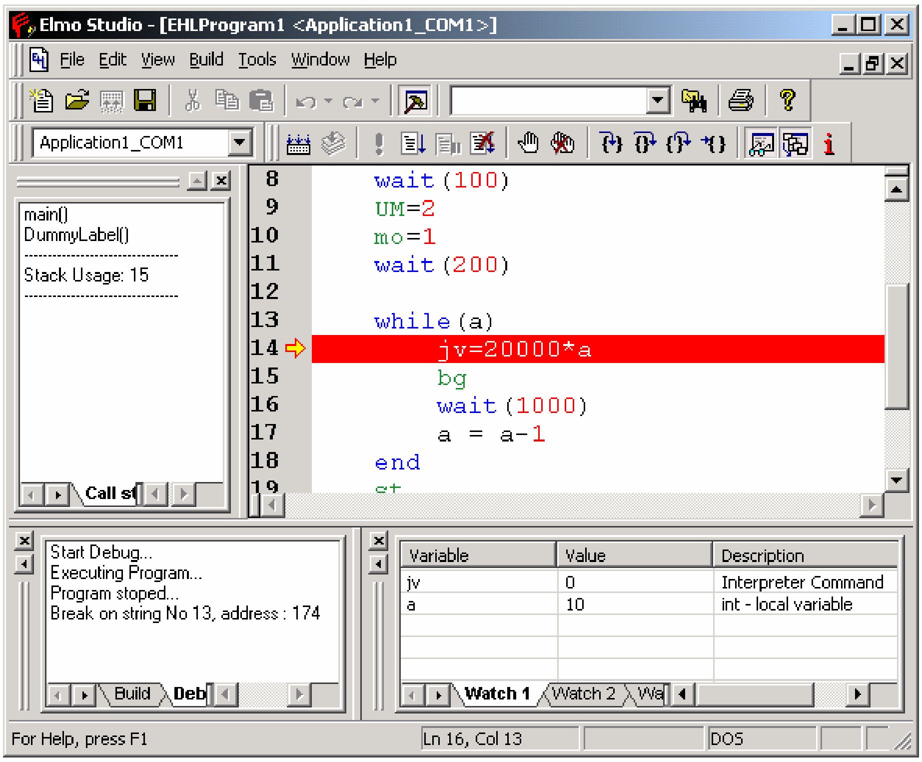


Figure 4-2: ***Debugging Process***

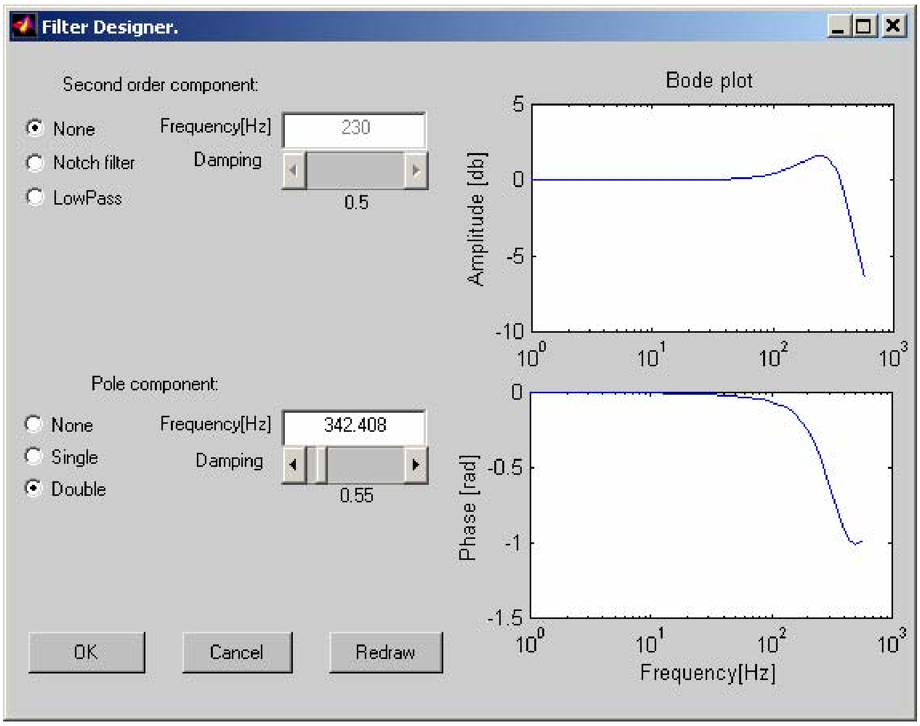
The following **debugging tools** are available for enabling you to manually debug your program in conjunction with the Elmo Studio debugger:

Button	Menu Option	Description
	Build - Debug - Break	Stop the debugger as it is running.
	Build - Debug - Step Into	Enter function and stop at first command.
	Build - Debug - Step Over	Execute next instruction and then stop.
	Build - Debug - Step Out	Continue and then stop at first instruction before current function is called.
	Build - Debug - Run to Cursor	Halt execution at source line at which the cursor is presently standing (no breakpoint needed).

Table 4-4: ***Debugging Tools***

Appendix: Using the Advanced Filter Designer

The Advanced Filter Designer is a tool that enables you to manually design an advanced filter by editing its parameters and viewing the filter transfer function. You access the Filter Designer by clicking **Advanced Filter Designer** in the Tuning Velocity Loop dialog box (section 2.8) or the Tuning Position Loop / Tuning Dual Loop dialog box (sections 2.9 and 2.10).



If the settings of the current filter are editable, you can modify the values using the controls in the Filter Designer dialog box in order to achieve an optimal filter.

The filter may consist of one or two components: **Second order component** and **Pole component** in the Filter Designer dialog box. You may enable or disable one or both of the components by selecting the **None** option.

For a notch filter, select **Notch filter** from the **Second order component** block. The following is the notch filter block formula:

$$\frac{s^2 + 2 \cdot d \cdot \omega \cdot s + \omega^2}{s^2 + \omega \cdot s + \omega^2}$$

The user-configurable parameters are:

- *d*: the **Damping** ratio
- $\omega = 2\pi f$: the notch **Frequency** (in Hz)

This block may be used to damp a single resonance of the system. Set the *f* value to equal the resonance frequency you wish to damp (in Hz) and change the damping to decrease the level of resonance inhibition. Decreasing the damping value widens the frequency range in which the filter is most active, and increases the inhibition. However, due to the uncertainty of resonances and certain drawbacks of notch filters, low damping factors should not be used. As a rule of thumb, always use a damping factor greater than 0.07; lower values may be dangerous.

For a low-pass filter, select **LowPass** from the **Second order component** block. The following is the low-pass filter formula:

$$\frac{\omega^2}{s^2 + 2 \cdot d \cdot \omega \cdot s + \omega^2} \qquad \omega = 2 \cdot \pi \cdot f$$

The user-configurable parameters are:

- *d*: the **Damping** ratio
- $\omega = 2 \pi f$: the complex pole **Frequency** (in Hz)

You can use this block to inhibit the system response to frequencies higher than *f*.

For a single pole component, select **Single pole** from the **Pole component** block. The following is the single-pole block formula:

$$\frac{\omega}{s + \omega}$$

The user-configurable parameter is:

- $\omega = 2 \pi f$: the pole **Frequency** (in Hz)

This block may be used to inhibit the system response to frequencies higher than *f*, as with a complex pole filter. This is probably a weaker solution than a complex pole because high frequencies are better attenuated by complex pole than by a pole.

For a double pole component, select **Double pole** from the **Pole component** block. The following is the double-pole component formula:

$$\frac{\omega^2}{s^2 + 2 \cdot d \cdot \omega \cdot s + \omega^2} \qquad \omega = 2 \cdot \pi \cdot f$$

This formula is the same as the **LowPass** formula in the **Second order component** block and therefore has the same parameters.

While designing the filter, you can click **Redraw** to view the Bode plot. This will be the Bode plot of the discrete results.

To accept the filter design and use it as the new filter, click **OK**. The **Advanced Filter ON** button will be turned on (green). To disable the filter, return to the Advanced Filter Designer and select **None** for each of the two components.

Glossary

Acceleration — The rate at which speed increases, in counts/sec^2.

Advanced controller — A controller with a more complex structure than a simple PI or PID. It can include several notch filters, low pass filter, poles and zeros.

Application continuous current — The maximum current, in amperes, that can be used by the specific application. This value must be equal to or less than the continuous stall current defined by the manufacturer for the selected motor.

Application peak current — The maximum short-term current, in amperes, to be used with the specific application during the design phase of auto-tuning. This value must be equal to or less than the peak current defined by the driver manufacturer.

Application speed limit — The maximum motor speed, in RPM or meters per second, used for the application. This value must be equal to or less than the maximum mechanical speed defined by the motor manufacturer.

Bandwidth — The difference, in hertz, between the highest and lowest level of a frequency range. In a standard feedback system, responses to reference commands at low frequencies are probably large; however, at high frequencies, they decrease. It is customary to look at the frequency where the response command amplitude drops to 70% in amplitude, compared to a low frequency response as a split frequency between good response and bad response. This frequency is used as a figure of merit for the system and is called the system bandwidth.

Baud rate — The rate at which digital data is transmitted, in bits per second.

Bounded motion — For auto-tuning purposes, a mode in which the motor motion is limited to movement around a selected fixed point. This mode is used with linear motors and with rotating motors whose shafts must remain within specified angle boundaries.

Coil length — The distance, in millimeters, that the motor travels during one electrical revolution. This value is normally listed in the motor datasheet.

Continuous stall current	The maximum continuous current, in amperes, allowed for the motor. This value is defined by the motor manufacturer.
Counts	The position unit of measurement of the drive. Four times the number of electronic pulses sent by an encoder in one revolution.
Deceleration	The rate at which speed decreases, in counts/sec^2.
Displacement	The change in position of the system – measured in encoder units – with respect to a specific reference point.
Driver continuous current	Same as continuous stall current.
Encoder magnetic pitch	See Magnetic pitch.
Encoder resolution	The length of the position unit in linear motors. Four times the encoder grating pitch value, because it takes into account two encoder slot transitions (high-to-low and low-to-high) and two sets of slots (A and B).
Free motion	The mode in which rotating motors operate without restriction, in terms of angle and position.
Free-wheel	A state in which power to the motor is turned off and the motor continues to rotate freely, by inertia.
Gain	The ratio of the output signal magnitude to the input signal magnitude.
Gain coefficients	Gain parameters used in the gain scheduling algorithm. See also KI and KP.
Gain scheduling	The means of accommodating for known variations in the dynamics of a system. Gain scheduling uses a customized algorithm that calculates adaptive gain modifications in order to improve system stability and accuracy.
KI	The integral gain parameter used to define a PID filter. It functions to reduce sharp peaks and to obtain a smoother step response.
KP	The proportional gain parameter used to define a PID filter. It functions as a means to obtain an optimal closed-loop transfer function.
Low pass	A filter whose transfer function transfers low frequencies and blocks high frequencies.
Magnetic pitch	The distance of one electrical cycle in a linear motor. See also Encoder resolution.

Maximum mechanical speed The maximum motor speed defined by the manufacturer, specified in m/sec for linear motors and RPM for rotating motors.

Notch filter A filter that blocks a defined band of frequencies and transfers all frequencies above and below that band. For example, a filter has a transfer function of the form:

$$\frac{s^2 + 2 \cdot \xi_1 \cdot \omega \cdot s + \omega^2}{s^2 + 2 \cdot \xi_2 \cdot \omega \cdot s + \omega^2}$$

where $\xi_1 < 0.5$ and $\xi_2 \approx 0.5$. Its Bode plot, depicted in Figure G-1, has a hole at frequency ω. The filter is therefore said to be a notch filter at frequency ω. The purpose of a notch filter is to block energy transfer to the motor around the notch frequency. This is one means of preventing a motor and load from accumulating energy at their resonance frequency, thereby avoiding shaking of the load.

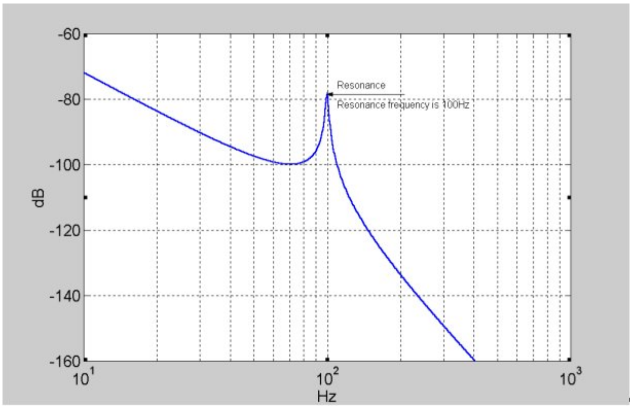


Figure G-1: **Bode Plot of a Notch Filter, d1=0.01, d2=0.5 Located at 100Hz**

Phase The angular relationship between voltage and current waveforms.

PTP Point-to-point motion, according to which the motor moves from its present position to a final point. The motor reaches the final point at zero speed and then remains at that point. The trajectory to the final point is calculated based on the speed, acceleration and deceleration limits.

PVT table Position-, velocity- and time-tabulated motioned defined in a table array.

Record resolution

The amount of time between consecutive sampling points. This value is calculated in conjunction with maximum recording time. With the Composer Wizard, the maximum number of recorded data points is 8000. The record time length is calculated as:
8000/(number of values) * record resolution.

Resonance

A condition whereby a large oscillatory amplitude occurs as a result of a small amplitude of periodic input, with a frequency close to one of the regular system frequencies. For example, a motor has a load whose transfer function has the Bode plot depicted in Figure G-2. It has a local maximum at the frequency 100 Hz; therefore, it can be said that the motor and load have a resonance frequency of 100 Hz. The systems absorbs energy at its resonance frequencies, thus tending to oscillate at those frequencies unless the controller is well designed to eliminate this phenomenon.

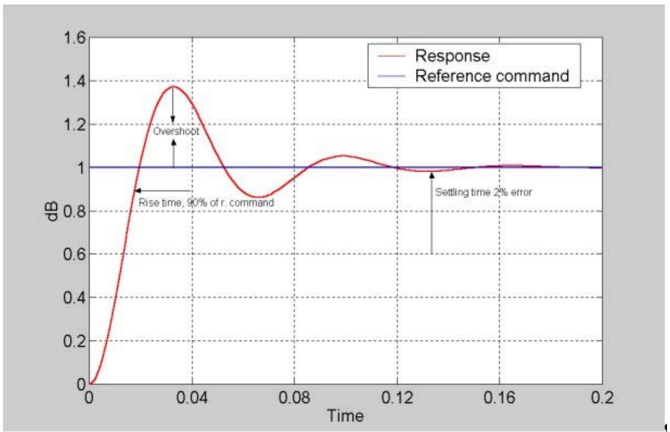


Figure G-2: **Example of a Bode Plot of a Transfer Function Including Resonance**

Smooth factor

The time, in milliseconds, that a motion speed profile is curved. The degree to which the "sharp corners" of a motion speed profile are curved. Smoothing a profile increases the time required to complete the motion.

Step response

The time required by a system for an output to pass through a specified percentage of a process. For example, a feedback system has a closed loop transfer function of:

$$\frac{\omega^2}{s^2 + 2 \cdot \xi \cdot \omega \cdot s + \omega^2}$$

The system responds to a step command of 0 up to time 0, then 1 at any positive time.

The following figure is an example in which ω = 100 and ξ = 0.3.

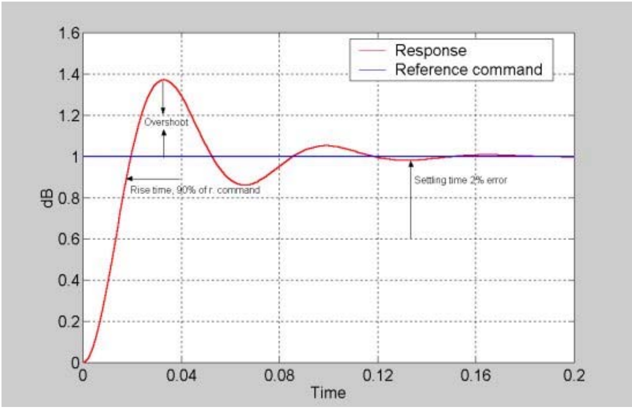


Figure G-3: **Figures of Merit in a Step Response**

Transfer function

A mathematical expression or a graph that expresses the relationship between the outgoing and the incoming signals of a process or control element. An important property of motors is that their response to a pure sinusoidal current signal is also sinusoidal at the same frequency. Suppose a pure sinusoidal current signal at frequency ω and amplitude A is injected into a motor, and the motor speed is sinusoidal with an amplitude B and phase φ relative to the current signal. The transfer function of the motor at the frequency ω therefore has an amplitude B/A and phase φ (Figure G-4).

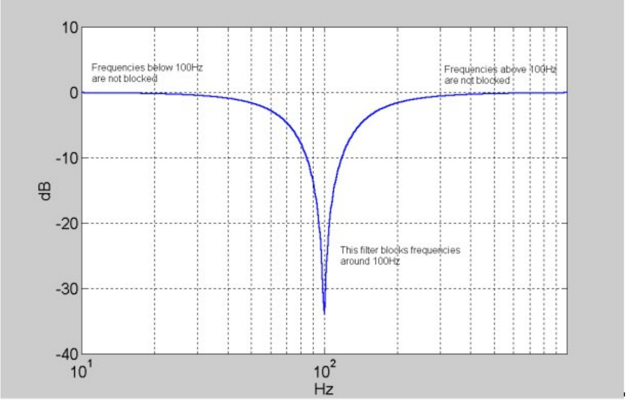


Figure G-4: **A Transfer Function with 1.66 Amplitude and -90° phase**

Trigger logic

A pulse that activates a function (either *high* or *low*).

Index

R

S

T

V

W

X