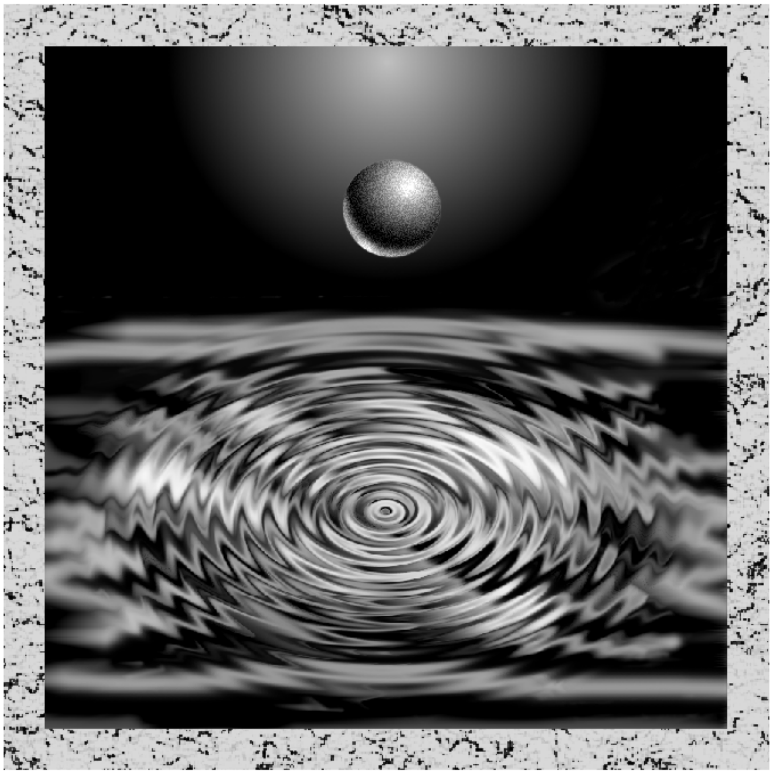
SigGenRP



User's Guide

TDT

.

SigGenRP User's Guide – Version 4.4

Copyright

Licenses and Trademarks

Printed in U. S. A.

Updated: 4/2/2009 11:26 AM

Contents

Preface

Software Philosophy

TDT's philosophy on software development is simple. We design comprehensive drivers that control all aspects of the hardware from common, high-level programming languages. TDT’s real-time Processor visual design studio (RPvds) enable scientists to produce high-quality signals tailored to meet their specific needs from a drag-and-drop environment. This approach works quite well in most research environments. With TDT's software, scientists no longer face the arduous task of programming each experiment in low-level machine code. Furthermore, they avoid the limitations imposed by "turn-key" software.

There has been an increasing demand for an interactive environment for designing complex stimuli for auditory experimentation. In response to this need, TDT has raised its approach to software development to a higher level by designing a powerful and flexible software package, *SigGenRP*. SigGenRP offers a signal generation environment that accesses the power of the System 3 hardware. It is a complete stimulus design application and the basis for a suite of TDT programs (including BioSigRP) specifically developed for auditory research.

SigGenRP is the most complete signal design package available today. Based on standard Windows techniques, SigGenRP is at once powerful and easy to use. Many complex auditory stimuli can be generated with ease when using SigGenRP. Principal investigators as well as students can quickly learn to use SigGenRP to design, specify, and preview signals. In addition to offering an extensive menu for generating waveforms, SigGenRP allows you to bring in your own waveforms, including sampled speech. This flexibility is useful, and often necessary, when designing complex signals.

As with all software applications, SigGenRP has its inherent limitations. Undoubtedly, some researchers find it necessary to design and generate highly sophisticated stimuli that are beyond the current capabilities of SigGenRP. For this reason, TDT still maintains as its highest priority continued support of its software drivers.

Organization of the Manual

The *SigGenRP User's Guide* is divided into three parts:

Part 1	SigGen Fundamentals
Part 2	Illustrative Examples
Part 3	Appendices

SigGen Fundamentals

Part 1, SigGen Fundamentals, presents all the basic SigGen concepts necessary to build a signal. Part 1 guides you through the process of building a SigGen signal and also serves as a general reference tool. Part 1 provides:

- General information about SigGenRP.
- Basic information about SigGenRP's features.
- A detailed explanation of the signal design and preview process.

General Information

General Information describes the purpose and uses of SigGenRP, SigGenRP installation, and basic SigGen concepts. General information is presented in:

- *Chapter 1* Introduction

Basic Information

Basic information about SigGenRP's features is described in the following section:

- *Chapter 2* Learning the Basics

The Signal Design Process

The steps required to design and preview a signal are presented below, along with their associated chapter.

Step	Chapter
1. Define signal parameters	*Chapter 3* Preparing to Build a Signal – The First Step
2. Define segment parameters	*Chapter 4* Signal Segments – The Building Blocks of a Signal
3. Define components	*Chapter 5* Components – The Building Blocks of Segments
4. View, modify, and/or preview completed segments	*Chapter 6* Working with Segments
5. View, modify, and/or preview completed signals	*Chapter 7* Working with the Complete Signal

Illustrative Examples

In Part 2, Illustrative Examples, various SigGenRP signals are designed in a step-by-step manner. These sample signals illustrate many powerful SigGenRP features. The illustrative examples were designed to:

- Provide general examples utilizing various SigGenRP features.
- Provide examples geared toward speech and auditory scientists.

General Examples

SigGenRP has numerous features that enable you to design complex signals with ease. Examples of signals designed with many of these features are presented in the following chapter.

- *Chapter 8* General Examples

Examples for the Speech Sciences

SigGenRP may be used to process speech signals, design vocal source signals, or even generate simple steady-state vowels. Such examples are provided in the following chapter:

- *Chapter 9* General Examples: Speech Sciences

Examples for the Auditory Sciences

SigGenRP provides numerous features that enable you to quickly and easily design complex auditory stimuli. Examples of common auditory stimuli are presented in the following chapter:

- *Chapter 10* General Examples: Auditory Sciences

Appendices

Part 3, Appendices, presents additional information in the following appendices:

- Appendix A SigGenRP Menus and Toolbars
- Appendix B Customizing the SigGenRP Display
- Appendix C Using SigGenRP with System II
- Appendix D Index Variable Example

Part

1

SigGen Fundamentals

Chapter 1 Introduction

Welcome to SigGenRP, the most complete Windows-based signal generation package available today.

What Is SigGenRP?

Signal generation is an integral part of virtually all tests and experiments involving acoustic stimuli. The SigGenRP stimulus design application enables scientists to quickly design, specify, and preview the complex auditory stimuli used in their experiments. It offers a fully interactive, high-level signal development environment that can be used to generate sophisticated signals to be used with the entire suite of TDT's run-time programs.

SigGenRP Capabilities

SigGenRP is a complete stimulus design environment with many desirable features. You can use the system to generate waveforms for saving to disk; however, when combined with a run-time application, which can be TDT’s BioSigRP, SigPlayRP, OpenEx, Brainware, or a user’s program developed with SigGen engine, SigGenRP becomes a comprehensive design platform where stimuli for all trials of an experiment are specified. SigGenRP files are also supported in TDT’s OpenEx software suite. SigGenRP provides control over the type of stimulus, frequency content, gating function, level, starting time, overall duration, and calibration. SigGenRP even provides a means for controlling the systematic variation of these parameters from one stimulus interval to another.

Calibration

SigGenRP allows you to specify all signal levels in terms of a calibrated reference level. This reference level can be specified as 0 dB SPL or some other reference.

Signal Design

SigGen allows you to design a wide variety of signals. Signal components can be specified in the time or frequency domains. You can choose between tones, clicks, noise, bands of noise, harmonics, FM sweeps, and waveforms from disk. SigGenRP's features enable you to build complex tones and amplitude modulated signals. You can also specify exact signal onset and offset times.

Playback Preview

SigGenRP allows you to simulate stimulus presentation for an entire experiment by playing through all of the signal intervals. The values of variables are displayed and are reflected in the real-time generation of the signals.

Displays

SigGenRP's two main displays are the Signal window and the Segment editor. The Signal window displays the overall stimulus signal, showing the placement of segment waveforms along a temporal axis. The Segment editor displays design parameters, the time-domain segment waveform generated by these design parameters, and the magnitude spectrum of the segment waveform. You can *zoom-in* on a specific area of the signal. Values associated with the displayed signal are viewed in an information window. You may place this window anywhere within the screen display.

SigGenRP will support any screen resolution currently used by your Windows system.

Hardware Support

SigGenRP supports TDT's System 3 or System II instrumentation. You can design simple, single channel stimuli or complex stimuli that can play to two channels simultaneously. SigGenRP also supports programmable attenuation through TDT's PA5s or PA4's (for System II). SigGenRP allows users to design complex stimulus presentation using RPvds, TDT's graphical interface for programming a signal processor.

When SigGenRP is used with the Real-time processors (RPx devices) the program uses a software emulation of the System II array processor to design signals. The generated waveform is then loaded to a memory buffer on an RCO circuit. This RCO circuit determines the maximum number of channels that are played out, the maximum sample frequency and also the triggering properties of the system. In addition, the RPx devices can be used in place of System II peripheral hardware such as waveform generators, cosine switches, and programmable filters.

SigGenRP continues to support any combination of TDT's System II programmable attenuators, waveform generators, cosine switches, and programmable filters. Information on using SigGenRP with System II can be found in Part 3 of this guide.

Who Can Use SigGenRP?

Anyone with a PC, Windows 2000 (or higher), and TDT System 3 or System II instrumentation can use SigGenRP. Signals designed with SigGenRP can be used in a variety of disciplines. Users include:

- Auditory scientists
- Speech scientists
- Physiologists
- Signal processing instructors

Auditory Scientists SigGenRP is an excellent tool for the custom design of complex auditory stimuli. SigGenRP provides a means for controlling the presentation of auditory stimuli across multiple intervals. Used in tandem, SigGenRP and TDT's BioSigRP application provide auditory scientists with the means to custom design an auditory stimulus; present multiple intervals; vary signal parameters across intervals; and collect, average, and process raw response data.

Speech scientists Speech scientists may use SigGenRP to design signals for a variety of tasks. With SigGenRP you can design a signal for use as the excitation source of a vocal tract model. Through the use of custom spectral shaping, you can generate steady-state vowels.

Physiologists SigGenRP is a very flexible signal design tool. Many types of stimulus signals may be designed with SigGenRP.

Signal Processing Instructors Because it is an intuitive, graphically based environment, SigGenRP is extremely useful as a tool for teaching students the fundamentals of digital signal synthesis.

How Does SigGenRP Work with System 3?

System 3 hardware and software make stimulus generation and presentation more flexible. SigGenRP takes advantage of System 3 hardware features in several ways. First, SigGen files can be used with a variety of hardware devices. Second, circuits can be customized for stimulus presentation.

System 3 RPx devices combine a programmable signal processor with memory, high quality DAC’s and ADC’s, and digital I/O. Users generate custom circuits for complex stimulus generation and data acquisition in RPvds, a graphical circuit design program. The circuits are compiled and saved so that they can be used with any TDT software. SigGenRP uses the circuit design to control stimulus presentation, sample period, and other processing abilities. A few System 3 hardware and software features are important to better understanding how to use the TDT software.

1) Depending on their capabilities, most RPx devices allow the user to set the sample period in discrete steps of 6K, 12K, 25K, up to 200kHz. System 3 RPx devices (with the exception of the RV8) can deliver excellent waveforms almost up to the Nyquist limit (.85x). Choose the sample rate that is 2.2x greater than the largest frequency.

2) SigGenRP comes with several preconfigured RCO circuit files. These circuits were designed for use with BioSigRP, although they can also be used with other TDT system software. Custom designed RCO files can also be designed for use with other TDT software. *(See RP Device Selection and Setup in Chapter 3 for more information on RCO files).*

How to Build and Present SigGenRP Signals

SigGenRP allows you to build a variety of complex signals. More importantly, however, SigGenRP allows you to specify how these signals will be presented as stimuli. This section answers two questions critical to the design and presentation of SigGenRP signals:

- How does SigGenRP build signals?
- What is the SigGen Index (SGI)?

How Does SigGenRP Build Signals?

SigGenRP builds signals in a hierarchical fashion.

- A SigGenRP signal consists of at least one segment.
- A segment consists of one to three components. Segments without components can also exist.

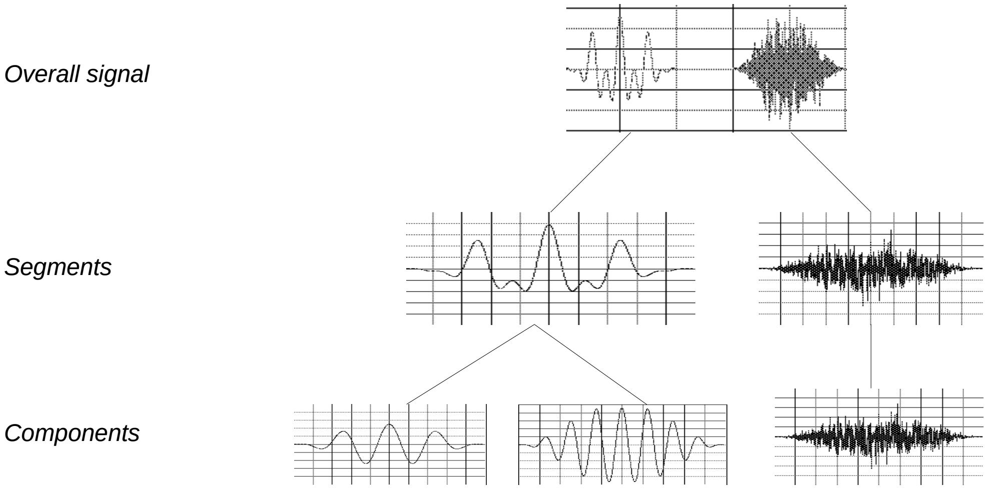
In order to build a signal with SigGenRP, all you have to do is specify the parameters of each signal, segment, and component.

Steps

1. Define "signal" parameters including: duration of the entire signal, devices to be used for playback, overall attenuation, calibration, amplitude and phase correction, and variables.
2. Define "segment" parameters including: segment duration, segment onset, gate, component generation method and application method, and segment level.
3. Define "component" parameters. There are various parameters to specify depending on the type of component to generate. These parameters may be in the time or frequency domains. It is these parameters that actually generate the component waveforms that comprise your signal.

Results

Defined components are additively combined to produce a segment. Segments may be added together or multiplied by each other to produce SigGen signals.



What is the SigGen Index (SGI)?

In most cases, a signal is built as the first step in a larger process. Often the signal will be presented as an auditory stimulus in an experiment or test. The stimulus may be presented over multiple intervals with signal parameters varied as a function of the interval.

The *SigGen Index* (SGI) is simply another name for the interval number. During stimulus presentation, signal parameters are varied as a function of the SigGen Index (SGI).

How to Use SigGen Signals

SigGenRP provides a simple, yet powerful, means for designing and previewing a wide variety of signals. Signal design, however, is often just the first step in a larger process. The real power of SigGenRP lies in its ability to generate files that can be used by other signal processing/experimental applications.

The Power of Signal Files

Signals designed in SigGenRP may be saved in formats that can be read by a variety of signal processing applications. SigGenRP signals may be saved in the following formats:

- SigGenRP format
- Binary file format

SigGenRP File Format

Files saved in the SigGenRP format are known as SigGen files. These files contain signal parameter information that can be used to generate SigGen signals. These files also contain the variable information necessary to control the presentation of stimulus signals across multiple intervals, or SigGen Indices (SGIs). These files can be read by many TDT software products. Assisted with SigGen Engine, you will be able to read the SigGenRP format for use in your own run-time programs written in a high level language. SigGen Files are saved with the default extension *.sig*.

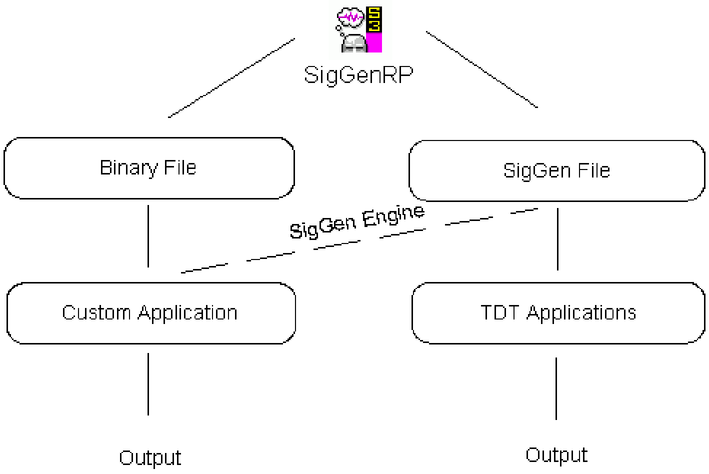
Binary File Format

Files saved in binary format may be easily used with many commercial application software packages or user’s programs. These files are saved in 16-bit integer binary format and consist of the waveform only. Default extension for binary signal files is *.16*.

Using SigGen Signals

The process of designing and using SigGen signals is as follows:

1. Design the signal with SigGen.
2. Save the signal to a file.
3. Use the file as input to a signal processing or data acquisition application.
4. Process output.



Applications

SigGenRP may be used by scientists and researchers from a variety of disciplines. Below are just a few examples of how SigGen signals may be used in the disciplines of speech and hearing science.

Hearing Sciences

SigGenRP may be used to generate complex auditory stimuli. When used together, SigGenRP and TDT's BioSigRP application can perform experiments designed to obtain auditory evoked potential data, including: auditory brainstem response, auditory middle latency response, late evoked potential, and electrocochleograph data. Below are just a few of the kinds of stimulus signals that can be generated with SigGenRP.

- Forward masking
- Backward masking
- Customized noise
- Amplitude modulation
- Tone bursts
- Clicks
- Tone sweeps

Speech Sciences

SigGenRP can be used to generate a number of signals for use in speech sciences. Such signals can be used in education, research, and vocal tract modeling.

- Clicks
- Noise
- Steady state vowels
- Excitation source for vocal tract models

SigGenRP can also be used to manipulate signals from speech files to generate new signals.

Before You Begin

With a bit of preparation, signal generation with SigGenRP is quick and easy.

What You Need

See your Microsoft Windows documentation.

- Windows fundamentals

  You should be comfortable with Windows basics: starting Windows; using the mouse; manipulating windows; opening, closing, and saving files.

- Signal processing
- A basic understanding of System 3 hardware and RPvds

  A basic knowledge of signal processing is necessary. You should understand the parameters necessary for specification of a signal in the frequency and time domains.

See Chapter 1, How to Build and Present SigGenRP Signals.

- Basic SigGen concepts

  You should recognize the terms *component*, *segment*, *signal*, and *SigGen Index*.

Installing the Software

Requirements (for use with System 3)

In order to run SigGenRP, you must have the following:

- Microsoft Windows 2000 or higher
- A monitor with at least VGA resolution graphics. Super VGA (1024 x 768) resolution graphics highly recommended
- TDT Drivers software (version 44 or greater)
- TDT's Real-time Processor and ZBUS hardware (required for playing back signals in SigGenRP)

Installation

To install SigGenRP

1. Make sure your TDT hardware is installed and functioning properly.

   Refer to the *System 3 Installation Guide*.

2. Insert the software CD in to your CD-ROM drive.
3. Click Install TDT drivers, click System 3, and click Install SigGenRP.

Chapter 2 Learning the Basics

Getting Started

Starting SigGenRP

To start SigGen

- ➢ Click the Start button on the Windows taskbar, point to Programs, point to TDT Sys 3, and click SigGenRP on the start menu.

  or

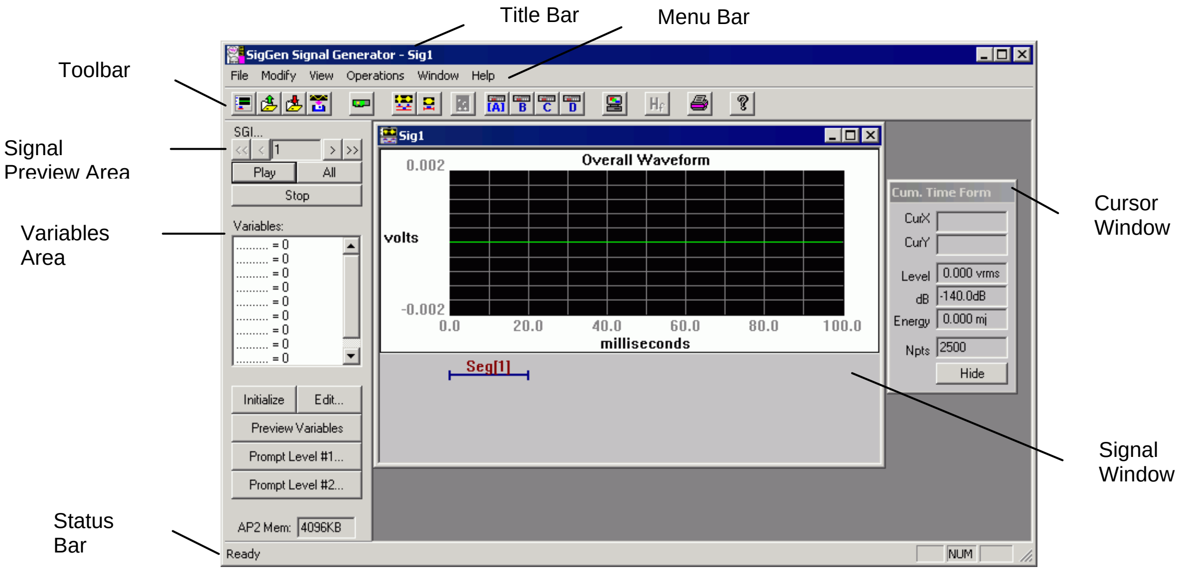
- ➢ Double-click the SigGenRP icon.

  The SigGenRP main window opens.

You are now ready to begin designing signals with SigGenRP.

Getting to Know the SigGen Main Window

Upon starting SigGenRP, you will be presented with the SigGenRP main window. It is from this window that you will be able to view the complete signal once you have defined all of its segments. Since SigGenRP is a MDI environment, one or more signals can be edited at the same time. The SigGenRP main window also provides functions that allow you to preview the signal through multiple intervals, or SigGen Indices.



SGI... 1

Variables: .... = 0 (×9)

Level 0.000 vrms

dB -140.0dB

Energy 0.000 mj

Npts 2500

AP2 Mem: 4096KB

Ready

The window contains the following sections:

Title Bar Displays "SigGen Signal Generator" and the name of the current SigGen file. If the signal has not been saved to a SigGen file, a default name, Sig1 for example, will be displayed.

Menu Bar Contains a list of menus used for building, manipulating, or saving your signal.

Toolbar Provides easy-to-use icons for the most commonly used SigGenRP menu commands.

Signal Window Displays the signal(s) as a time-domain waveform. More than one signal can be displayed at the same time, but only one can be the current signal (the one you are working on).

Signal Preview Area Allows you to preview the stimulus through one specific interval or all intervals.

Variables Area Variables allow you to specify the manner in which signal parameters will be varied from one interval to another. This area displays the variable names and their current values corresponding to the current SGI.

Cursor Window Provides a readout of parameter values pertaining to any selected segment or signal.

Status Bar Provides you with information concerning the selected command.

Using the SigGen Menus and Toolbar

SigGenRP provides you with a full set of menu commands and a toolbar buttons. Using these menus, commands, and toolbar buttons, you can create, open, and save signals; build new signals or edit existing signals; print signals; and make choices about your screen display. The SigGenRP menus and toolbars are covered in detail in the Appendix A.

The Signal Window

Signals are displayed in child windows known as signal windows.

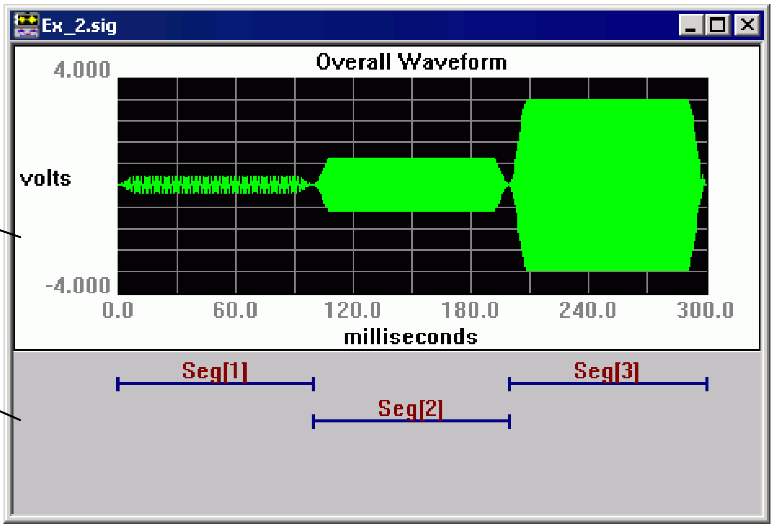
To open a new signal window

➢ Choose New from the File menu.

or

Click on the New Signal icon on the toolbar.

More than one signal can be generated, displayed and manipulated concurrently under SigGenRP. Each signal has its own Signal Window.



Signal Area

Segment Area

Each Signal Window consists of:

Signal Area Contains the time-domain waveform display.

Segments Area Displays graphic representations (segment bars) of the durations and temporal positions of each segment in the signal. At any time scale, up to four overlapping segment bars can be displayed.

SigGen builds a signal from mini-signals called segments. When you are designing signals in SigGen, you are actually designing at least one segment, specifying how these segments are related to each other in time, and specifying how the segments will affect the entire signal.

Because each signal must contain at least one segment, "new" signals, by default, contain one segment. This default segment is defined as follows:

- Segment onset is 0 milliseconds.
- Segment length is 20 milliseconds.
- No components are defined in the segment.

Opening a SigGen Signal

You can always open an existing SigGen signal file to modify it again, or to view it. When an existing SigGen file is opened in SigGen or an application, SigGen will reset SigGen Index to 1 and load all necessary files (including the normalization file, variable files and signal component files; whichever apply).

Note: SigGen or the application loading a signal will look for needed files in the path(s) specified in the SigGen file. An error will be displayed when a file cannot be found or loaded, and an incorrect signal will be generated. You must remember that when you move a normalization file, variable file, or a signal component file to another location or delete it from its current location, related SigGen files should be updated to reflect the change.

To open an existing signal

1. Choose Open Signal... from the File menu, or click on the Open Signal icon on the toolbar to open the Open File dialog box.
2. Click on or type in the desired file name.
3. Click the OK button.

Note: The selected signal file must be in SigGen format. These files are typically saved with the extension, *.sig*. Trying to open a non-SigGen file will generate an error and crash the software.

To open a recently updated signal

1. Choose the File menu.

   SigGen displays the names of the last 4 signal files saved at the bottom of the file menu.
2. Select the desired file name.

Importing a Non-SigGen Signal

You may import non-SigGen waveforms into SigGen. SigGen signals are built from segments. Segments, in turn, are built from components. When importing a non-SigGen signal, you are actually using this signal as a SigGen component. This means that all non-SigGen signals must be imported as components. The procedure for importing components is described in the Importing Components section of Chapter 5.

Working with Multiple Signals

Because SigGenRP supports the Windows Multiple Document Interface (MDI), it is possible to be working with more than one signal window at a time. SigGenRP supports common Windows features for working with multiple windows.

Saving Signals

Signals may be saved in the following formats:

- SigGenRP format

  Files saved in the SigGenRP format contain signal, segment, and variable parameter information that can be used to generate the signal through multiple intervals, or SigGen Indices (SGIs). These files can be read by all TDT application software products. Another TDT software product, SigGen Engine will allow you to use SigGen files with user applications. The default extension for files saved in SigGenRP format is *.sig*.

- Binary file format

  You can also save signals as time-domain waveforms stored in 16-bit integer binary file format. These files are given the default extension of *.16* unless you change it to another one. Since such files are raw data files containing only the current waveform, they do not include any other information in the SigGen files, and each interval must be saved separately. You must display the waveform of another interval before you save that interval. Raw data files can be used as a SigGenRP component, or with many commercial programs and your own run-time programs.

Saving Files in SigGenRP Format

To save a new file in SigGen format

1. Select the signal by clicking anywhere in the signal window.
2. Choose Save... or Save As... from the File menu, or click the Save Signal icon on the toolbar.

   The File Save As dialog box will be displayed. You may assign a name to your new signal. If you do not give an extension, SigGen will use the default *.sig* extension. It is best to use the default *.sig* file extension.

To save an existing file

1. Select the signal by clicking anywhere in the signal window.
2. Choose Save... from the File menu

   The file will be saved with its current name.

Saving Files in Binary Format

To save a file in binary format

1. Select the signal by clicking anywhere in the signal window.
2. Choose Save Wave... from the File menu.

   The Save Wave File dialog box will be displayed. You may assign a name to your new signal. If you do not give an extension, SigGen will use default .16 extension. It is best to use the default *.16* file extension.

   The displayed waveform will be saved as a 16-bit binary file.

The Signal Design Process

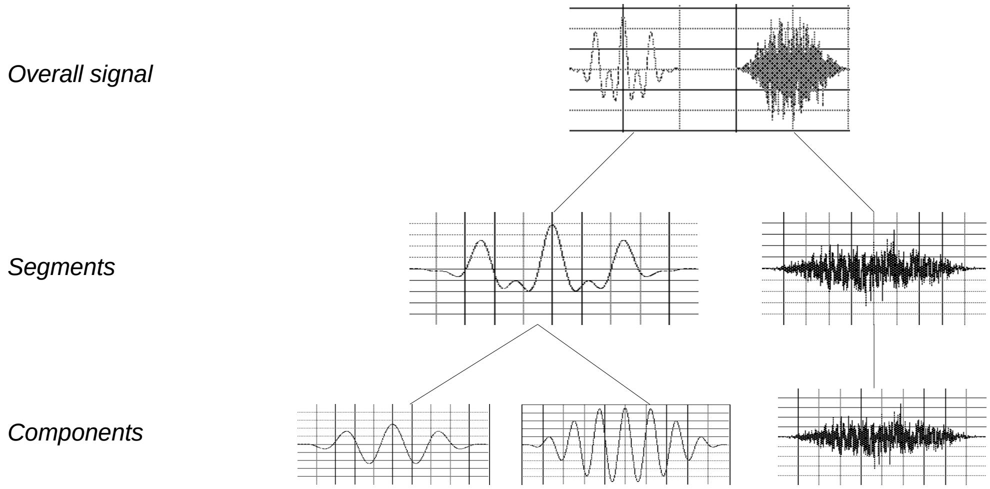
SigGenRP builds signals in a hierarchical fashion.

- A SigGen signal consists of at least one segment.
- A segment consists of up to three components. A segment with zero components is also acceptable.

In the process of building a signal with SigGenRP, you have to either specify the parameters of each signal, segment, and component, or accept their default values. To build complicated signals, you also have to define various variables and/or create necessary file(s).

Results

Defined components are additively combined to produce a segment. Each segment may be added or multiplied to the signal to produce a final SigGen signal.



In the chapters that follow, the details of this process are explained. The steps required to design and preview a signal are presented below, along with their associated chapter. You may find it more helpful to go over some or all of the illustrative examples in the second part of this guide.

Step	**Chapter**
1. Define signal parameters	*Chapter 3* Preparing to Build a Signal – The First Step
2. Define segment parameters	*Chapter 4* Signal Segments – The Building Blocks of a Signal
3. Define components	*Chapter 5* Components – The Building Blocks of Segments
4. View, modify, and/or preview completed segments	*Chapter 6* Working with Segments
5. View, modify, and/or preview completed signals	*Chapter 7* Working with the Complete Signal

Chapter 3 Preparing to Build a Signal - The First Step

Prior to designing the constituent segments of a signal, you should define the parameters that will determine the entire signal, as well as any variables that might be used to change signal, segment, or component parameters as a function of the presentation interval, or SigGen Index (SGI).

RP Device Selection and Setup

RP device selection and setup includes:

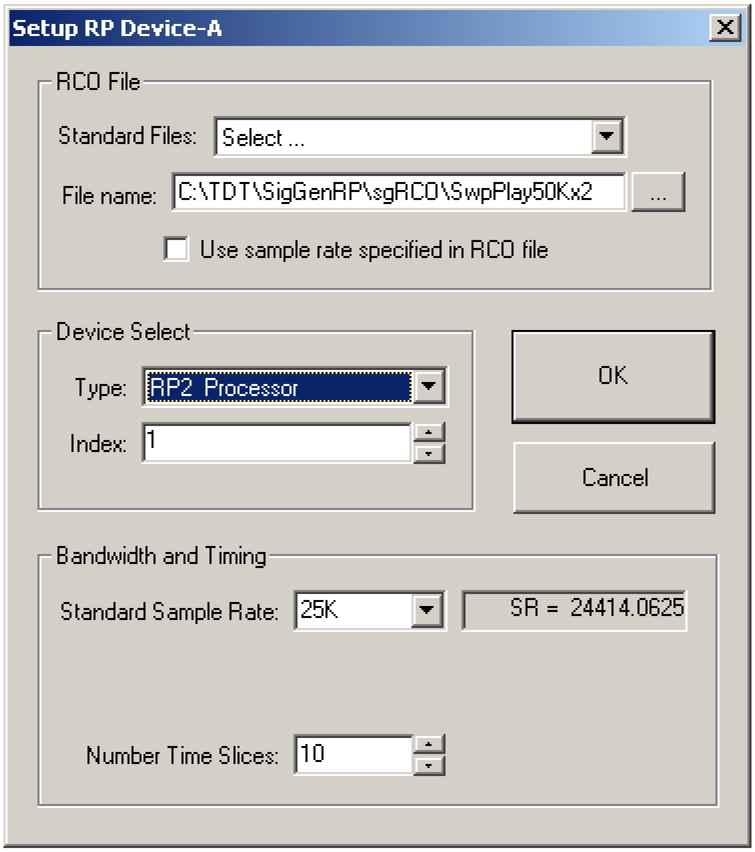
- Device selection
- Logical number of device
- Type of stimulus playback circuit
- Sample Period of the device

RP device parameters are defined from the Setup RP Device dialog box, which can be accessed from the Modify menu or from the RP Device icons on the toolbar.

To access the RP device dialog box

➢ Click RP Devices… from the Modify menu, or click the RP Device [A] button “A” on the toolbar.

The RP device setup dialog box has three sections. The first section selects the RCO file to use with the playback device. The second section selects the device and module. The third section controls the sample frequency.



Setup RP Device-A

RCO File

Standard Files: Select ...

File name: C:\TDT\SigGenRP\sgRCO\SwpPlay50Kx2

□ Use sample rate specified in RCO file

Device Select

Type: RP2 Processor

Index: 1

Bandwidth and Timing

Standard Sample Rate: 25K

SR = 24414.0625

Number Time Slices: 10

RCO file

Users can choose from the RCO files that come with SigGenRP or make their own. Standard files come in two modes, sweep play and continuous play. Under most circumstances users will use sweep-based presentation. The maximum sample frequency for each RCO file is given in parenthesis. Clearing the Use sample rate specified in RCO file check box can modify the sample rate of the RCO file.

Deciding Which RCO File To Use

For more information about RPx and RCO files see the RPx Help.

There are two types of standard RCO files, sweep and continuous. Sweep RCO files generate a new signal based on the duration of the signal, i.e. the interstimulus interval, and continuous RCO files repeat the signal for the length of the stimulus cycle. Sweep RCO files are good for Pyschoacoustic testing and Physiological recordings. Continuous RCO files are excellent for analyzing room acoustics and Distortion Product Otoacoustic Emissions (DPOAEs).

In addition RCO files differ in the number of output channels (1-4) and the maximum sample rate of the circuit. The maximum sample rate of the circuit determines the maximum stimulus frequency. The greater that value of the sample rate the greater the stimulus frequency that can be used. For example, a 50kHz SigGenRP RCO file can generate stimuli up to 22 kHz, which is just a little less than the nyquist limit. The sigma-delta parts on the RP2.1, RA16, and RL2 produce smooth signals with no aliasing up to the nyquist limit.

Device Select

Under Device Select, users must select the device type (RP2.x, Medusa, Stingray, or Barracuda) and the index number.

Type

All RPx modules are accessible from the dialog box. The RP2, RP2.1, and the Stingray, each have two DAC outputs while the Medusa has eight DAC outputs and the Barracuda can have up to eight DAC outputs. The standard RCO files are device independent so the standard RCO files will work with any of the System 3 devices.

Index

Many TDT workstations include multiple RPx modules. To select the RPx module for stimulus presentation use the devices index. Devices are numbered logically beginning with one. A device index of zero means that the module type has not yet been selected. Logical device numbers may not correspond to the physical device organization. To ensure that the devices are ordered consistently each time the system is used make sure that devices are turned on in the same order each time the system is turned on. To check that the devices have been properly selected run zBUSmon to check the order of the logical devices.

Bandwidth and Timing

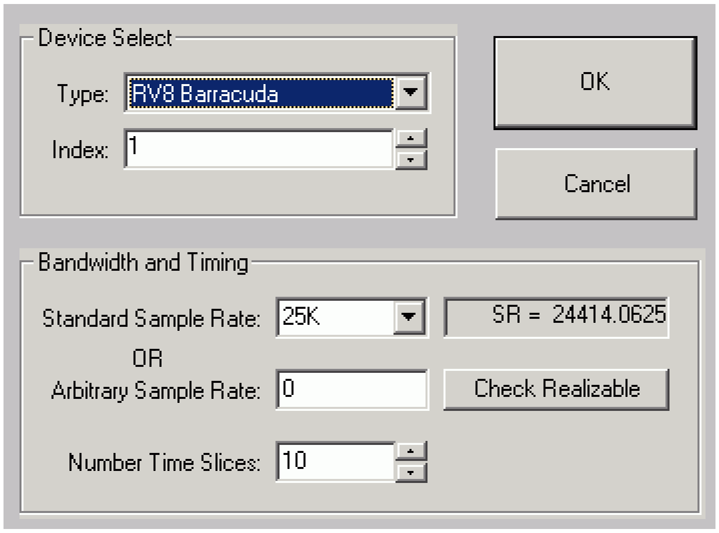
The Bandwidth and Timing settings are only visible when the Use sample rate specified in RCO file check box has been cleared. Under most conditions you can select one of four or five sample rates. In addition, systems with the Barracuda can access its arbitrary sample rate.

Standard Sample Rate

When the Use sample rate specified in RCO file check box has been cleared the standard sample rate of the RCO file can be modified by selecting a new rate from the drop-down list. System 3 devices can use sample frequencies of 6, 12, 25, 50 and in some cases 100, 200 and 400 kHz.

The sample frequencies are approximate. The true sample frequency is shown in the text box to the right. An RCO file may not be able to operate at some of the higher sample rates. For the standard files, users can select the maximum sample rate. For user generated files users should determine the maximum sample rate by checking the cycle usage of the circuit.

Arbitrary Sample Rate



Type: RV8 Barracuda

Index: 1

Standard Sample Rate: 25K

SR = 24414.0625

Arbitrary Sample Rate: 0

Number Time Slices: 10

The Arbitrary Sample Rate field is only accessible when the Barracuda device is selected as the device type. The Barracuda can have arbitrary sample rates from 10 Hz to 500000 Hz. Any RCO file can be modified to use these arbitrary sample rates. Again it is necessary to check the cycle usage to insure that the circuit will work at the higher sample rates. The sample frequency is calculated from a 0.04 microsecond clock. Press the check realizable button to determine the true sample rate.

Signal Setup

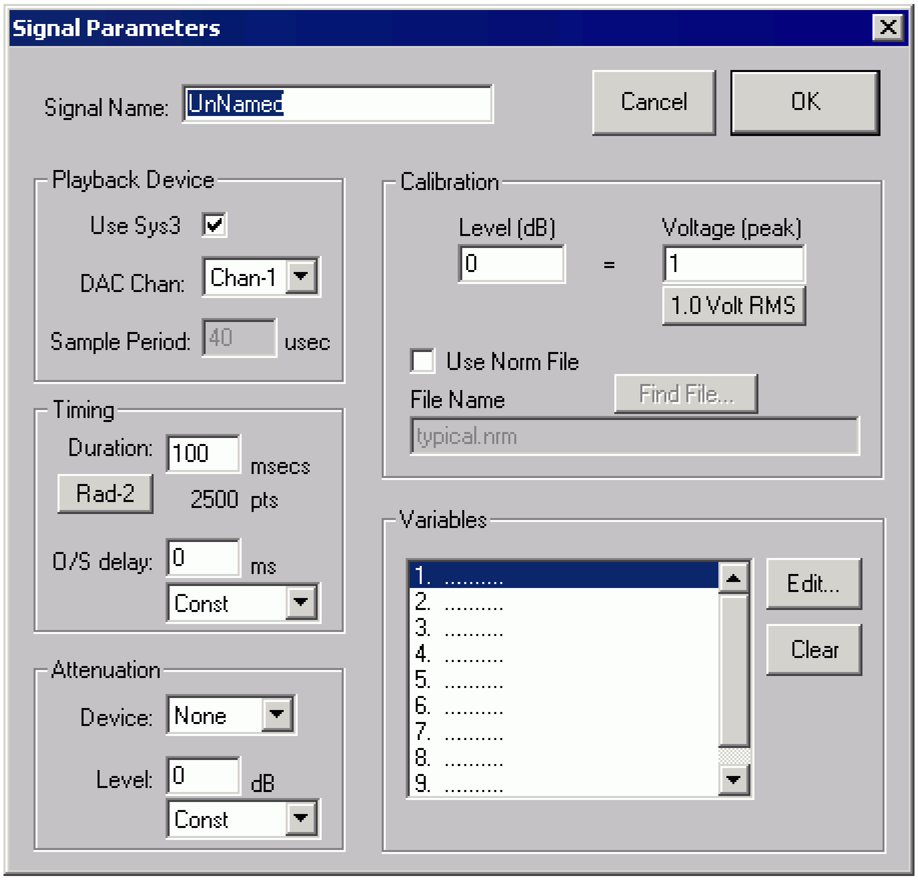
Overall signal parameters include:

- A name for the signal
- The duration of the signal
- The sample period of the time-domain waveform (for System II)
- The playback D/A channel
- Any external attenuation
- Calibration specifications
- Variable definitions

Signal parameters are defined from the Signal Parameters dialog box, which can be accessed from the Modify menu, or from the Modify Signal icon on the toolbar.

To access the Signal Parameters dialog box

➢ Click Signal... from the Modify menu, or click on the Modify Signal icon on the toolbar.



Signal Name: UnNamed

Use Sys3 ☑

DAC Chan: Chan-1

Sample Period: 40 usec

Duration: 100 msecs

2500 pts

O/S delay: 0 ms

Const

Device: None

Level: 0 dB

Const

Level (dB) 0 = Voltage (peak) 1

□ Use Norm File

File Name typical.nrm

1. ....
2. ....
3. ....
4. ....
5. ....
6. ....
7. ....
8. ....
9. ....

Signal Name

Signals are defined using names that may not exceed 15 characters in length. Signal names can be used to identify a specific signal in applications.

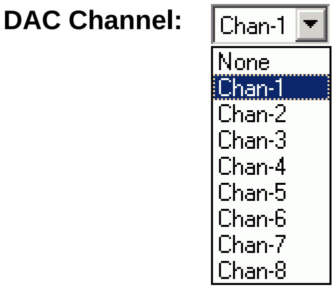
Playback Device

Use Sys 3

SigGenRp continues to support TDT’s System II hardware platform. By default SigGenRP is configured for System 3 and the Use Sys 3 check box should be selected. For more information on using System II see *Appendix C, Using SigGenRP with System II*.

DAC Channel

SigGenRP supports up to eight D/A channels for signal definition, and one channel for signal preview. Prior to signal preview, you must select the playback channel.



DAC Channel: Chan-1

All possible D/A channels are displayed in the D/A Channel box. Your system may not have all the channels displayed in the D/A Channel box. Check your D/A hardware to find the number of D/A channels available, and which one you want the signal to be played to.

To select/deselect a D/A Channel for signal preview

- Click the desired channel.

Run-time applications, such as TDT's BioSigRP, read and use the D/A channel information stored in the SigGenRP file

When your device uses channel numbers higher than eight

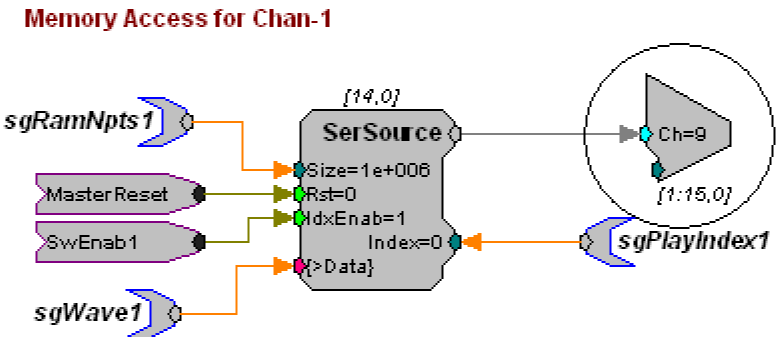
System 3 processors such as the RX6, RX8, and RZ2 are configured with D/A channel numbers higher than eight.

When using RCO files designed for the RX6 the channel numbers have been shifted, allowing the D/A channels to be selected as Chan-1 and Chan-2.

The RX8 and RZ2 allow for custom D/A channel configurations. Refer to your system specs to determine the correct D/A channel values. Any RCO circuits must be modified accordingly.

For example:

1. Load the standard RCO C:\TDT\SigGenRP\sgRCO\SwpPlay50Kx2.rpd.
2. Locate the DacOut components in the circuit file – circled below. Double-click the component to change the channel value as needed. The diagram below would play the SigGenRP signal on D/A channel 9 when Chan-1 is selected in the SigGenRP DAC Chan drop down menu.

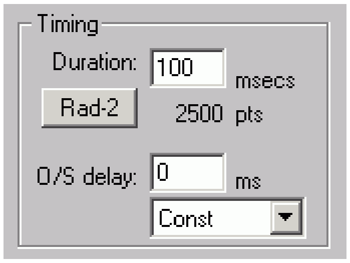


3. Save the newly created configuration with a different filename.

Sample Period

Sample period indicates the length of time in microseconds between successive data points in the time-domain waveform. With System 3 the sample period is directly tied to the Real-time processing device available in your system and the sample period box is unavailable in the Signal Parameters dialog box. The sample period is configured in the Setup RP Device dialog box. For more info see *RP Device Selection and Setup* at the beginning of this chapter.

Timing


Timing
Duration: 100 msecs
Rad-2
2500 pts
O/S delay: 0 ms
Const

Duration

The Duration field of the Signal Parameters box defines the total duration of the signal in milliseconds. The value entered in this field determines the scaling of the Signal Window x-axis.

Radix-2

Sometimes a duration corresponding to a radix-2 number of sampling points is desired. For example, to generate a waveform as short as possible while collecting enough points for Fast Fourier Transformation (FFT) analysis. In order to eliminate the calculation to get a duration with desired radix-2 samples, SigGen provides a quick function button that truncates the value in the Duration field so that the total number of points in the signal is rounded to the closest radix-2 number.

To truncate to radix-2

1. Enter the approximate duration in the Duration field.
2. Click Rad-2.

The duration will be recalculated so that the total number of points is defined in terms of radix-2. The new duration and number of data points will be displayed. If you feel a different radix-2 number is what you want, you can double the duration value (or divide it by 2), and then click the Rad-2 button to get the next radix-2 number.

O/S Delay

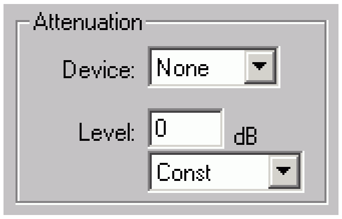
The O/S Delay field specifies the onset delay of the signal in milliseconds. Onset delay is defined as the delay between the initiation of an SGI and the beginning of signal presentation.

Attenuation

When signals are presented to TDT D/A devices, only 16-bit integer samples will be converted to analog signals. As a result, small amplitude signals will lose their details and appear distorted, and the signal-noise ratio will be reduced. To make a good use of the dynamic range of the D/A devices, you should make your signal’s amplitude as large as software and hardware allow, and use attenuators to adjust the signal level after D/A conversion to produce desired signal level.

When using a programmable attenuator, it is necessary to specify two parameters:

- Device
- Level



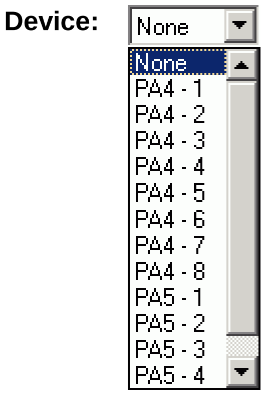
Attenuation

Device: None

Level: 0 dB

Const

Device



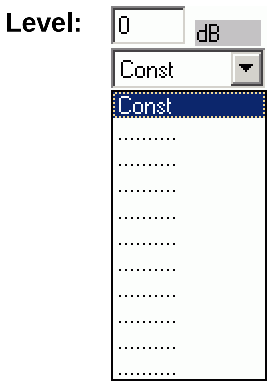
A list of attenuators is displayed in the Device box. Your system may not have all of the displayed attenuators available.

To select/deselect an attenuator

- Click the desired attenuator name or click None.

Level

The level field specifies the level of attenuation sent to a programmable attenuator.



Level: 0 dB

Const

Constant or Variable Level By default, Level is defined as Const, and therefore remains constant, regardless of the current SigGen Index value. Level may be varied as a function of the current SigGen Index value by assigning a predefined variable (see Creating Variables). Whether or not you can use a variable for Level depends on the run-time program.

To set Level to a constant or a predefined variable

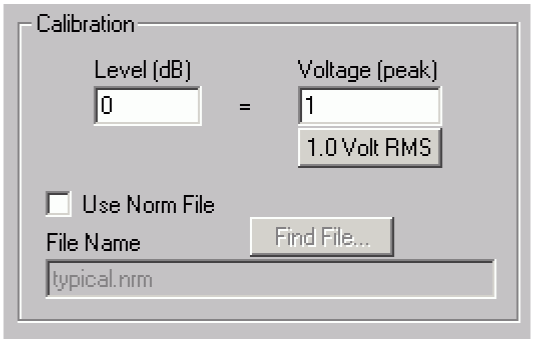
- Click Const or the desired variable name in the Level box.

If Level is a constant,

- Enter the value in the Level field.

Calibration

In designing signals in SigGen, a decibel (dB) value is used to specify a level of a segment gain, or a component level. Since all signals will eventually be presented in terms of volt, a dB-volt relation or calibration must be correctly entered. By default, all measurements of level in SigGen are calculated in dB volts with a reference of 1 volt peak. You may calibrate measurements of level by setting a specific voltage equal to a designated dB level, or using one Volt RMS equal to the dB level.



Calibration

Level (dB): 0 = Voltage (peak): 1

typical.nrm

Level

The Level field specifies a level in dB that equals the specified number of volts (peak), or one volt RMS.

Voltage

The Voltage field specifies the number of volts of the signal peak amplitude that equal the specified number of level units (dB). For example, in the specification (90 dB = 1 Volt (peak)), a tone that is specified to be 90 dB will produce a tone with a 1-volt peak amplitude.

One Volt RMS

Clicking this button sets Voltage (peak) field equal to 1 volt RMS (1.41421 (peak) volts). This is convenient when you want to make your calibration in terms of RMS values.

Example: ***Calibrating units to reflect SPL***

You may wish to calibrate SigGenRP's level units so that they correspond to the sound pressure level (SPL) output of your system.

To calibrate SigGenRP

1. Generate a pure tone signal of a specified number of peak volts.
2. Set Voltage equal to this number of peak volts.
3. Measure the output of your system in SPL.
4. Set Level equal to the measured output in SPL.

Normalization Files

There may be some instances when you may wish to modify the magnitude and/or phase of various frequency components in the SigGenRP signal. For example, if you want to generate a pure tone simulating the output from a speaker which has a nonlinear frequency characteristics. This can be accomplished through the use of Normalization Curves. The Normalization Curves are inputted to SigGenRP through a Normalization file, and will apply to the signal components generated in the frequency domain only.

Note: Normalization files may be applied to signals created in the frequency domain only. A normalization file does not affect a time domain signal.

What Is a Normalization File?

A Normalization file is an ASCII text file that normalizes a SigGenRP signal by specifying correction factors for magnitude and phase at specific frequencies. Parameters specified include:

- Frequency in Hz
- Magnitude correction in dB
- Phase correction in degrees

Creating a Normalization File

Normalization Files may be created with any editor that can save files in ASCII text format.

Default Naming Conventions. By default, SigGenRP looks for files with the *.nrm* file extension. It is a good idea to use this file extension when creating variable files.

Normalization File Structure

Each line in the Normalization file contains three values specifying the magnitude and phase correction factors for a specific frequency. The syntax is as follows:

frequency in Hz, magnitude correction in dB, phase correction in degrees

A comma "," is the required separator between two values in a line. An error will be generated in SigGen when one or more commas are missing from the normalization file.

An example Normalization file is shown below.

```
0.0, 0.0, 0.0
500.0, 0.0, -30.0
1000.0, -6.0, -60.0
2000.0, -12.0, -90.0
4000.0, -24.0, -120.0
```

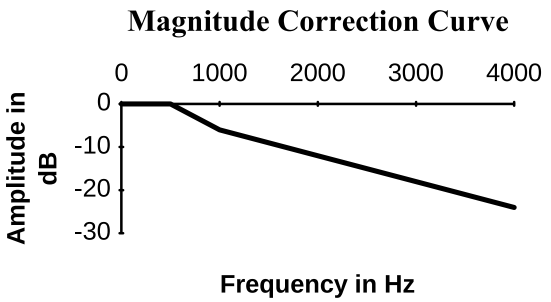
How Normalization Files Work

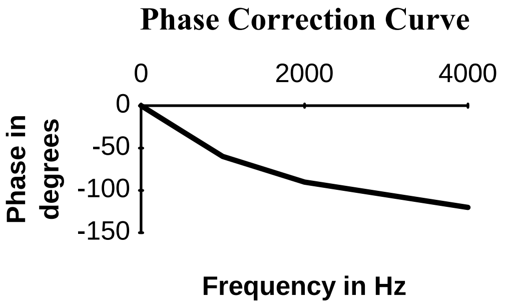
SigGenRP normalizes signals at the segment level. The frequency characteristics of each segment are adjusted according to the correction factors specified in the magnitude and phase correction curves. Specifically, magnitude and phase correction factors will be generated to every sample point of the segment, and added to the magnitude and phase parts of the segment. Consequently, segments generated using Time method will not be corrected by the normalization curves.

For example, the above Normalization file specifies the following correction factors:

	Correction Factor	
Frequency in Hz	**Magnitude in dB**	**Phase in degrees**
0	0	0
500	0	-30
1000	-6	-60
2000	-12	-90
4000	-24	-120

By using linear interpolation to calculate intermediate values, the following correction curves are produced.


Magnitude Correction Curve


Phase Correction Curve

Using a Normalization File

In order to use a Normalization file, you must:

- Indicate that you wish to use a normalization file.
- Specify the file name.

To use a Normalization file

- Check the Use Norm File check box.

To specify a Normalization file

- Type in the desired file name in the File Name field.

  or

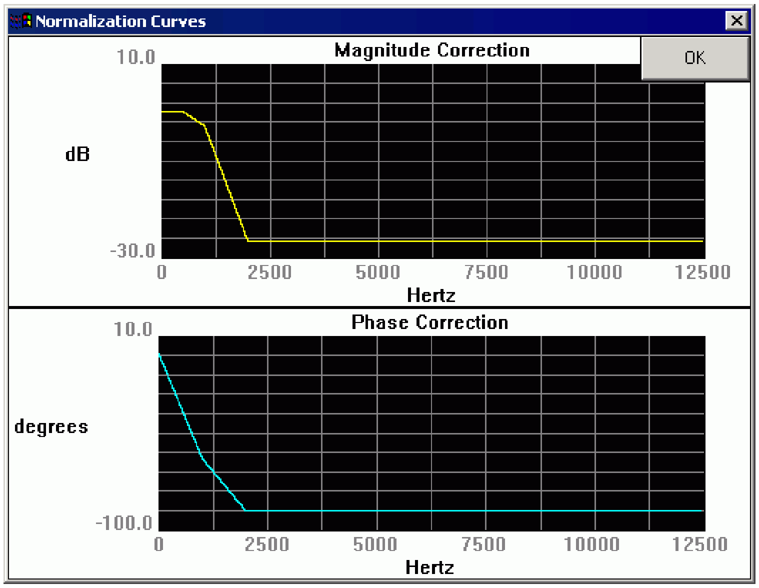
1. Click Find File.
2. Select the desired file.

Viewing Normalization Curves

From the SigGenRP main window, magnitude and phase normalization curves may be viewed in the Normalization Curves box.

To display the Normalization Curves box

- Select Norm Curve from the View menu or click on the Normalization Curves icon on the toolbar.



The Normalization Curves box displays:

- The Magnitude Correction curve
- The Phase Correction curve

Magnitude Correction Curve The Magnitude Correction curve plots magnitude correction in dB as a function of frequency in Hz.

Phase Correction Curve The Phase Correction curve plots phase correction in degrees as a function of frequency in Hz.

To hide the Normalization Curves box

- Click the OK button in the upper right corner.

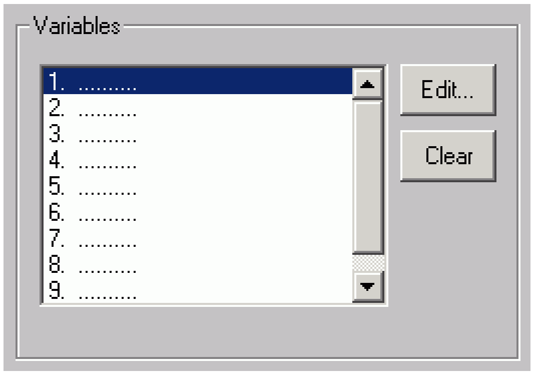
Variables

SigGenRP uses SigGen Index (SGI) and variables to generate waveforms of different signal intervals. SigGenRP allows you to specify variables. These variables may be used to change signal or segment parameters as a function of the presentation interval, or SigGen Index (SGI).

Up to ten variables can be defined in each signal. Variable values are functions of SigGen Index, and can be altered systematically, read in from a file, or specified by you prior to each interval.

Variables may be created, edited, and cleared in the Signal Parameters dialog box or from the Variables Area of the main window.

The Variables Area of the Signal Parameters Dialog Box



Edit...

A new variable (except an alternating variable) can be defined at any available place of the Variables box.

To create a new variable

For more information about creating variables, see Creating Variables in this chapter.

1. Click on an available variable place (..........) in the Variables box.
2. Click the Edit... button.

or

➢ Double-click on the available variable (..........) in the Variables box.

This will open the Signal Variables dialog box.

To edit an existing variable

1. Click the desired variable in the Variables box.
2. Click the Edit... button.

or

➢ Double-click the variable in the Variables box.

This will open the Signal Variables dialog box.

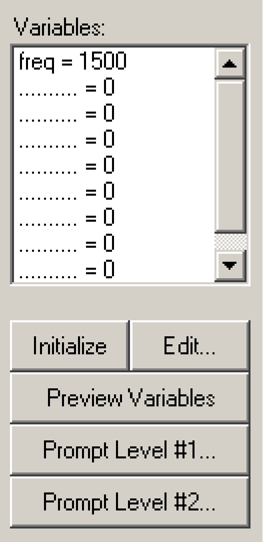
Clear

To delete an existing variable

1. Click on the desired variable in the Variables box.
2. Click the Clear button.

Note: When a variable is deleted, SigGenRP will remove its name from the Variables box, but will not always reset the parameters corresponding to this variable to their default values. Any parameter which uses this variable will still point to the place of this variable, and use zero as the variable value. Consequently, when a new variable is defined at this variable place, those parameters which use the deleted variable will automatically use the new variable instead. You should always remember that when you clear a variable, you must change all the signal/segment/component parameters which use the variable.

The Variables Group Box in the Main Window



Variables:

```
freq = 1500
.......... = 0
.......... = 0
.......... = 0
.......... = 0
.......... = 0
.......... = 0
.......... = 0
.......... = 0
```

Variables Box

For more information about creating variables, see Creating Variables in this chapter.

The Variables box displays a list of all currently defined variables along with their current values corresponding to the current SGI value.

When more than one signal is used in an application, the SigGen Index of the channel-1 signal will be the active SGI in the application. As a result, when two signals are created and used in an applications like BioSigRP, variables used in the signal assigned to the second channel should also be defined at the same variable locations in the first signal.

Initialize

Clicking the Initialize button sets the SigGen Index (SGI) back to 1, causing any variables associated with the current signal to be initialized to their starting values. Clicking Initialize also causes any variable files to be read. When using a variable file, it is a good idea to initialize the variable(s) prior to assigning the variable(s) to parameters.

Edit...

Clicking the Edit... button will open the Signal Variables dialog box. Clicking any variable or empty variable position will select that variable or position. When the Edit... button is clicked, the selected variable (or position) will be edited (or created) in the Signal Variable dialog box. Clicking Edit.. without selecting any variable or position will cause the first variable to be edited/created. Variables may be deleted in Signal Parameters dialog box.

To edit an existing variable

1. Click the desired variable in the Variables box.
2. Click the Edit... button.

   This will open the Signal Variable dialog box and you can edit the parameters of this variable.

To create a new variable

1. Click on one of the available variable positions (.......... = 0) in the Variables group box.
2. Click the Edit... button.

   This will open the Signal Variable dialog box. You can assign a name to this variable and select other parameters.

Preview Variables

Once you have defined the signal variables, it is possible to preview how they will change across intervals. Clicking the Preview Variables button opens the Variable Preview dialog box.

Prompt Level #1...

You may wish to specify the value of a variable on the fly during the presentation of a stimulus signal. From the Signal Variable dialog box, you may create user-prompts. When you save your SigGen signal file (see Working with Signals, Saving Signals), user-prompts will be saved as well. When using SigGen files to generate signals in other TDT applications, these user-prompts will be available.

When a user-prompt variable is defined, you can click on Prompt Level #1 or Prompt Level #2 button to access the Input Request dialog box. You may enter a new value for this variable. Current user-prompt variable value(s) will be used for all intervals of the signal preview.

To define Prompt Level #1

1. Click on the Prompt Level #1... button to open Input Request dialog box.

   If you have not defined a level 1 prompt, the Prompt Level #1... button will be disabled.
2. Enter the desired value in the Unknown Prompt field.

From the Input Request dialog, you can view the prompt message and enter any test value you may wish to use for signal preview.

Prompt Level #2...

In some cases, you may wish to vary parameters in a nested fashion. For example, you may wish to vary the frequency of a pure tone. For each frequency, you may in turn wish to vary the level.

You may specify a user prompt for a nested variable by using Prompt Level #2. In our example, this would be specified for level.

Note: Current versions of TDT's signal processing applications do not support nested variables or Prompt Level #2.

To define Prompt Level #2

1. Click on the Prompt Level #2... button to open Input Request dialog box.

   If you have not defined a level 2 prompt, the Prompt Level #2... button will be disabled.

2. Enter the desired value in the Unknown Prompt field of the Input Request dialog box.

If you have more than one Prompt Level #1 or Prompt Level #2 variable defined, Input Request dialog box will appear repeatedly until enough values are entered for all Prompt Level #1 or Prompt Level #2 variables.

Creating Variables

You may change the value of signal and segment parameters as a function of the SigGen Index (SGI) value by defining variables. From the Signal Variable dialog box you can define the names of up to ten variables, as well as specify starting values, minimum values, maximum values, and the manner in which the value will be varied.

You may access the Signal Variable dialog box from the following:

- The Variables area of the main SigGenRP window.
- The Variables group box of the Signal Parameters dialog box.

To access the Signal Variable dialog box

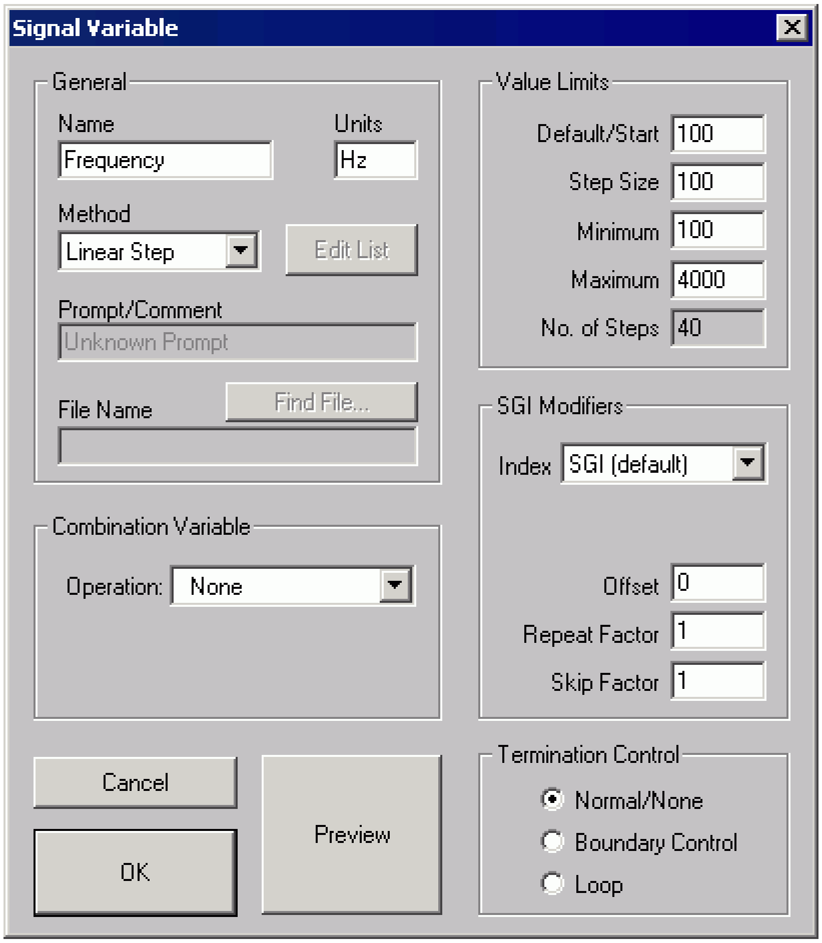
1. Do one of the following:

   ➢ Click a variable in the Variables box to edit an existing variable.

   or

   ➢ Click an available variable place (..........) in the Variables box to create a new variable.

2. Click the Edit... button.



Name: Frequency

Units: Hz

Method: Linear Step

Prompt/Comment: Unknown Prompt

Operation: None

Default/Start: 100

Step Size: 100

Minimum: 100

Maximum: 4000

No. of Steps: 40

Index: SGI (default)

Offset: 0

Repeat Factor: 1

Skip Factor: 1

● Normal/None
○ Boundary Control
○ Loop

General Variable Parameters

General variable parameters include:

- Name
- Units
- Method

Name

Variables are defined using names that may not exceed 12 characters in length. You can enter a variable name by typing in the Name field. When you change a variable name or its parameters, SigGenRP will also make changes to those parameters using this variable when editing variable is completed. Using a name reflecting the physical nature of the variable is highly recommended.

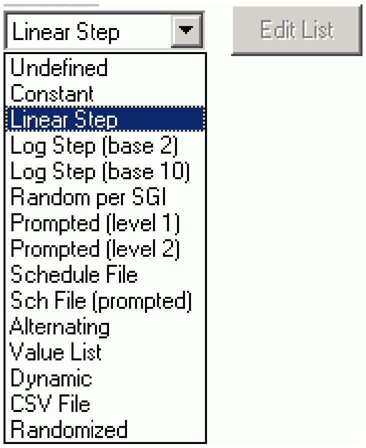
Units

In the Units field, you may enter up to 5 characters to *describe* the variable's intended unit of measure. The unit value you entered here is only a reminder, not a standard for data conversion. The unit consistency of your experimental system depends on your application program and the hardware you are using. Three basic units used in SigGenRP are: decibel (dB) for level, millisecond (ms) for duration, and Hertz (Hz) for frequency. You should use a correct unit for this field.

Note: The actual unit of measure assigned to the variable will be determined by the SigGen parameter with which the variable is associated. For example, you may have defined a variable named "Freq" and described its Units as "Hz." However, if you assign "Freq" to a duration parameter, its unit will be milliseconds.

Method

Method defines the mechanism (function) of altering variable values as a function of the SigGen Index (SGI). There are 14 methods available in SigGen.



Variable methods include:

- Undefined

  Undefined means that no Method has been defined for the current signal variable. The variable will be zero for all the signal intervals.

- Constant

  This method sets the variable equal to a constant value. This method is most useful when more than one SigGen parameter will be set to the same value. Constant defines a variable that will always have the same value as given in the Default/Start field of the Value Limits box. This value will be a floating-point constant between 1.2E-38 to 3.4E+38.

- Linear Step

  For each SigGen Index (SGI) increment, Linear Step defines a variable that will increment by the Step value specified in the Value Limits box.

- Log Step (base 2)

  Log Step (base 2) defines a variable that will perform as a base 2 exponential function of SGI:

  $$\text{Variable} = \text{Start x } 2^{\text{Step x (SGI-1)}}$$

  The Start and Step values are defined in the Value Limits group box of the Signal Variable dialog box.

- Log Step (base 10)

  Log Step (base 10) defines a variable that will perform as a base 10 exponential function of SGI:

  $$\text{Variable} = \text{Start x } 10^{\text{Step x (SGI-1)}}$$

  The Start and Step values are defined in the Value Limits group box of the Signal Variable dialog box.

- Random per SGI

  The value used for the variable is a random number between the Minimum and Maximum set in the Value Limits group box. The random numbers are generated with a uniform distribution within an SGI.

  A Step Size of zero will give a full floating-point random value within the specified range. Use a non-zero to define the resolution of the result (e.g.: a Step Size of 0.1 might produce values of 3.7, 2.9, 7.2, etc.).

- Prompted (level 1)

  To specify the value of a variable on the fly during the presentation of a stimulus signal, you may create user prompts. These prompts will be available in TDT's application programs.

  If you have defined a level 1 prompt for the selected variable, the Prompt Level 1 button will be enabled, and you can use this button to enter variable values when you preview the signal.

- Prompted (level 2)

  Using this method, you can be prompted for a value at run-time. Level 2 prompts are used for 2nd level nested variables. (Not currently implemented.)

  Variables defined with Prompt (level 2) method is not currently supported in TDT's application programs.

See Using Variable Files later in this chapter.

- Schedule File

  Variable values are read from a text file. The file name is specified in SigGenRP and is used by TDT's application programs.

  Each variable in a variable file can have up to 100 values.

- Sch File (prompted)

  Variable values are read from a variable file. The file name is specified by you during run-time. One usage of this method is to allow user to use more user-defined variable values. For example, when you want to use more than 100 specific variable values, you can put them into more than one variable file, and access them one after another.

  File prompt will be available for applications supporting this feature. While previewing signals in SigGenRP, you can display this prompt and assign variable file(s) by clicking Prompt Level #1 button in SigGenRP Signal Window.

- Alternating

  The value of the variable alternates between the values defined in the Minimum and Maximum fields. The variable value does not depend on SGI value, but on the number of the continuous presentation of the current signal interval. In the Variables group box, an alternating variable must be defined next to another predefined variable, or as the first variable.

  Note: Due to the nature of this variable, this feature is not available during Play or Play All in SigGenRP since the "continuous presentation of the current signal interval" condition is not met. However, you can test the alternating effect by repeatedly initializing variables. Alternating variables function when the SigGenRP signal is used as a stimulus in TDT's BioSigRP application.

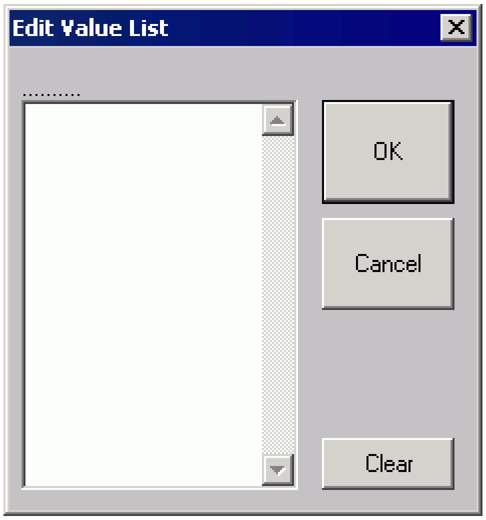
See the BioSigRP User's Guide for more information.

- Value list

  Variable values are specified in SigGen by you during the signal design process. A maximum of 100 variable values can be specified for the variable.

To enter or modify a variable list

1. Choose Value List from the Method box.
2. Click the Edit List button.
3. Enter values to define the variable.



See the <u>BioSigRP User's Guide</u> for more information.

- Dynamic

  Dynamic variable values are controlled by the run-time application, usually based on a subject feedback.

- CSV File

  The values for this variable type are reads from windows standard Comma Separated Value (CSV) files which allows for matrix (two dimensional) arrays where each row is an SGI and each column is a repeat. The CSV files used by this variable type are easily generated and saved using EXCEL or similar programs. A typical CSV file is shown below.

```
1, 6,11,16,21,26
2, 7,12,17,22,27
3, 8,13,18,23,28
4, 9,14,19,24,29
5,10,15,20,25,30
6,11,16,21,26,31
7,12,17,22,27,32
8,13,18,23,28,33
9,14,19,24,29,34
```

  A variable using this CSV file would use the current SGI to index down to a specific row. For example, for SGI = 6 the row with numbers 6,11,16,21… would be indexed. If this variable was used in the component of a repeating segment, the first repeat would use the value 6 and the second would use 11 and third 16 etc. This allows the user to get full control of how repeats are computed and where they are applied to the overall waveform

- Randomize

  The value used for the variable is a random number between the Minimum and Maximum set in the Value Limits group box. The random numbers are generated with a uniform distribution.

A Step Size of zero will give a full floating-point random value within the specified range. Use a non-zero to define the resolution of the result (e.g.: a Step Size of 0.1 might produce values of 3.7, 2.9, 7.2, etc.).

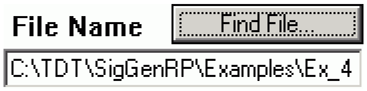
Prompt/Comment

If the defined method is Prompt (level 1 or level 2), you will be defining the variable values. If the defined method is File (prompted), you will be defining the name of the variable files. In these cases, a user prompt will be necessary, and the text string(s) entered in the Prompt/Comment field will be used as the Prompt. A prompt may be up to 50 characters in length including spaces.

To define a Prompt/Comment

1. Select Prompted (level 1 or level 2) or File (prompted) from the Method dialog box.
2. Type in the desired text in the Prompt/Comment field.

File Name



C:\TDT\SigGenRP\Examples\Ex_4

If the chosen method is Schedule File, a variable file must be specified.

To define a variable file

➢ Type in the file path and name

or

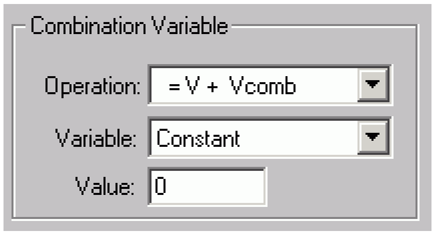
1. Click the Find File... button.
2. Choose the file name.

When you choose File (Prompted), you do not need to enter a file name at design time, however, you may need to supply the file when you run an application. A file for a File (Prompted) variable is not required when you preview a signal in SigGenRP.

Combination Variables

In many cases, a simple variable is not sufficient to define a signal. For example, consider a signal consisting of two segments in which the first segment has a variable duration, and the second segment starts 10 milliseconds after the first segment. You could calculate the start times of the second segment by hand, or you could define a combination variable.

A combination variable is a variable whose value is calculated by performing a mathematical operation on another variable or a constant.



Combination Variable

Operation: = V + Vcomb

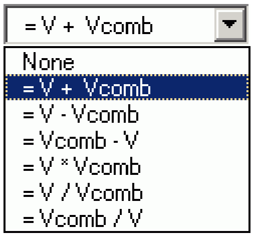
Variable: Constant

Value: 0

To Define a Combination Variable

1. Open the Signal Variable dialog, either by choosing Signal under the Modify menu and then clicking the Edit button, or by clicking the Edit button on the side bar.
2. Name and define the combination variable in the General box.
3. In the Combination Variable box select an Operation to apply to this variable (**V** in the Operation formula). Combination variables can be defined using addition, subtraction, multiplication, and division.

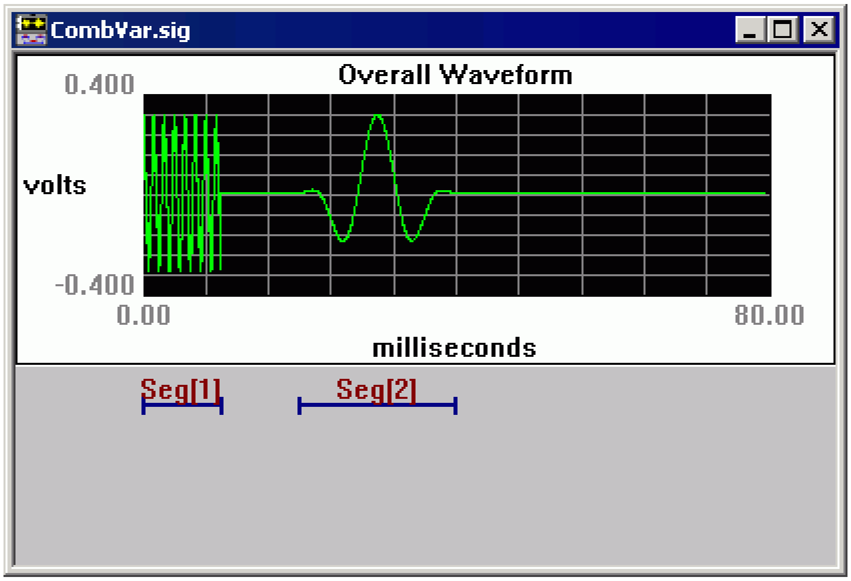
Operation:



4. The Variable drop down menu in the Combination Variable box, is used to define **Vcomb** in the operation. **Vcomb** can be defined as either a constant, and its value entered in the Value box, or it can be defined as another variable.

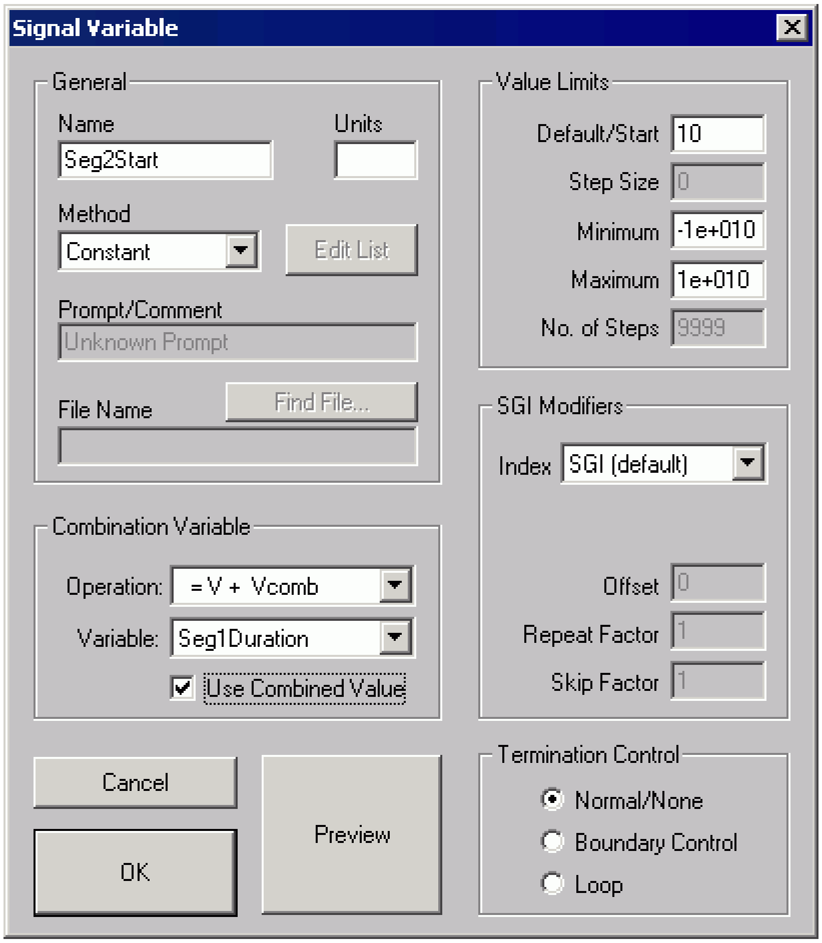
Combination Variable Example

This example illustrates how to use a combination variable to build a signal consisting of two segments in which the first segment has a variable duration, and the second segment starts 10 milliseconds after the first segment.



Seg1Duration is a simple variable that specifies the duration of the first segment. A combination variable, *Seg2start*, is defined so that segment 2 starts 10 msec after segment 1. *Seg2start* is defined with method Constant and Default=10.

In the Combination Variable box **Vcomb** is set equal to the variable *Seg1Duration*. **V+Vcomb** is selected so that *Seg2start* = 10 msec + *Seg1Duration*.



Signal Variable

General

Name: Seg2Start

Units:

Method: Constant

Prompt/Comment: Unknown Prompt

File Name:

Combination Variable

Operation: = V + Vcomb

Variable: Seg1Duration

☑ Use Combined Value

Value Limits

Default/Start: 10

Step Size: 0

Minimum: -1e+010

Maximum: 1e+010

No. of Steps: 9999

SGI Modifiers

Index: SGI (default)

Offset: 0

Repeat Factor: 1

Skip Factor: 1

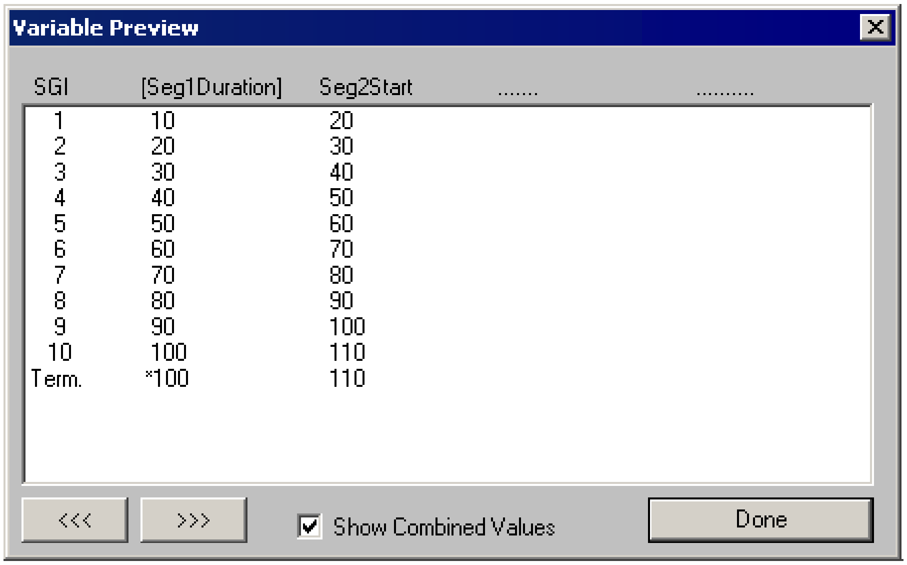
Termination Control

● Normal/None

○ Boundary Control

○ Loop

We can preview the results of the combination variable operation by selecting Preview. The uncombined value of the combination variable *Seg2start* is displayed. To see the results of the combination operation, check the Show Combined Values box. In this case you can see that *Seg2start* begins 10 ms after *Seg1Duration*.



Variable Preview

SGI	[Seg1Duration]	Seg2Start	....	....
1	10	20		
2	20	30		
3	30	40		
4	40	50		
5	50	60		
6	60	70		
7	70	80		
8	80	90		
9	90	100		
10	100	110		
Term.	*100	110		

☑ Show Combined Values

Combination Variables in BioSigRP

When SigGenRP signals that contain combination variables are used in BioSigRP, the combination variable value is calculated and used in generating the signal and controlling the peripherals. However, the screen and output files display the uncombined value of the variable. Thus, for the example shown above, *Seg2start* would be shown as 10 ms, even though it would change with different SGIs.

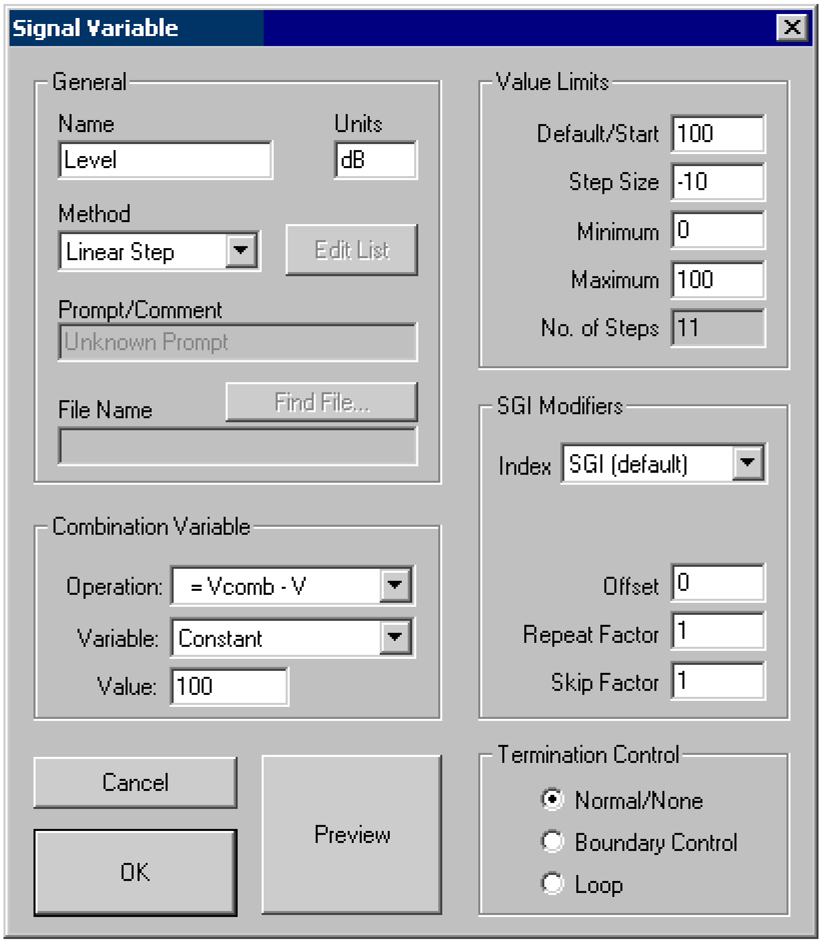
Example: Controlling SPL using a PA5

This comes in handy when controlling the level of signal presentation using a Programmable Attenuator (PA5). For example, a variable *Level* can be defined that steps through levels of sound presentation from 100 to 0 dB in 10 dB steps. However, to get 100 dB SPL, may require 0 dB attenuation, while a 60 dB SPL would require 40 dB attenuation. To do this define *Level* as a combination variable that is calculated by subtracting *Level* from a constant value (100 dB).

Operation:

Vcomb-V

where, **Vcomb**=100 (constant value) and **V**=*Level*



Signal Variable

General

Name: Level

Units: dB

Method: Linear Step

Prompt/Comment: Unknown Prompt

File Name:

Combination Variable

Operation: = Vcomb - V

Variable: Constant

Value: 100

Value Limits

Default/Start: 100

Step Size: -10

Minimum: 0

Maximum: 100

No. of Steps: 11

SGI Modifiers

Index: SGI (default)

Offset: 0

Repeat Factor: 1

Skip Factor: 1

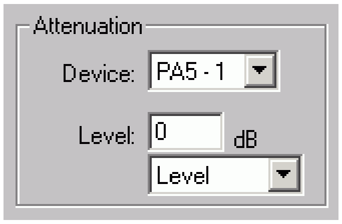
Termination Control

● Normal/None

○ Boundary Control

○ Loop

Then, for the PA5 use *Level* as the variable.



Attenuation

Device: PA5 - 1

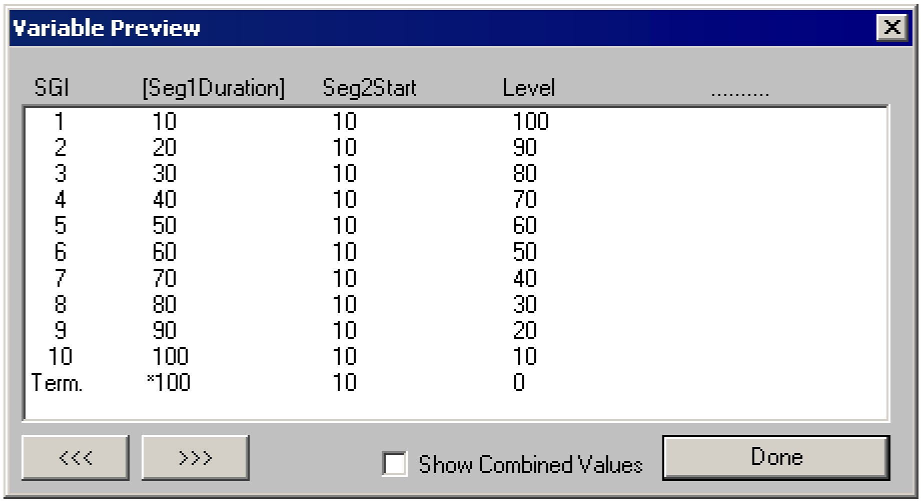
Level: 0 dB

Level

The uncombined values will show the true level of the sound. The combined values will show the attenuation applied to the PA5.

In BioSigRP, the PA5 will be loaded with the combined value, while the screen displays the uncombined value of *Level*. For a sound level of 100 dB, the PA5 will be set to 0 dB attenuation, and for a sound level of 60 dB, the PA5 will be set to 40 dB attenuation. You can see this in SigGenRP by stepping through each SGI, playing the signal, and looking at the attenuation loaded on the PA5.

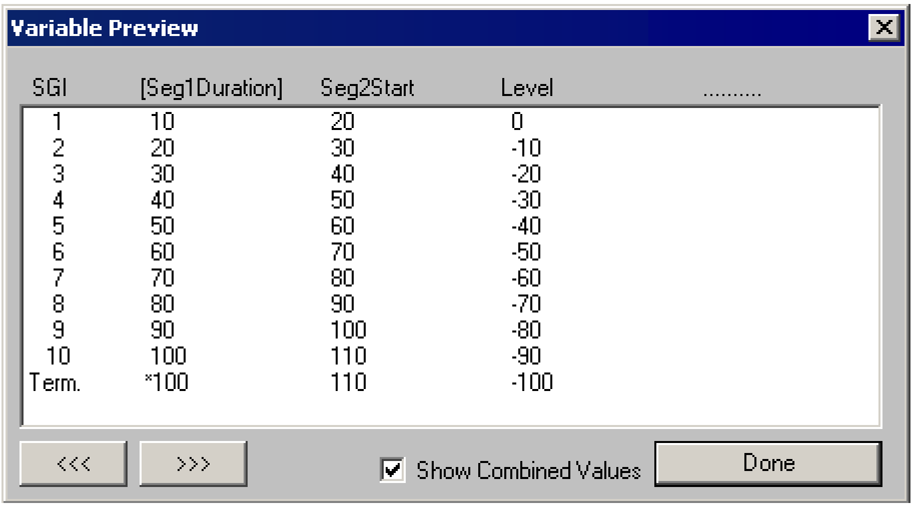
Uncombined Values



SGI	[Seg1Duration]	Seg2Start	Level	....
1	10	10	100	
2	20	10	90	
3	30	10	80	
4	40	10	70	
5	50	10	60	
6	60	10	50	
7	70	10	40	
8	80	10	30	
9	90	10	20	
10	100	10	10	
Term.	*100	10	0	

□ Show Combined Values

Combined Values

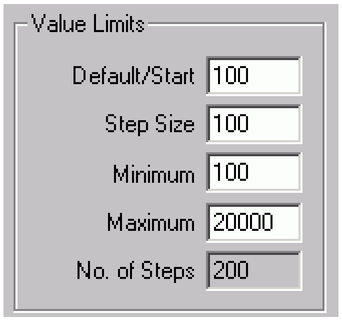


SGI	[Seg1Duration]	Seg2Start	Level	....
1	10	20	0	
2	20	30	-10	
3	30	40	-20	
4	40	50	-30	
5	50	60	-40	
6	60	70	-50	
7	70	80	-60	
8	80	90	-70	
9	90	100	-80	
10	100	110	-90	
Term.	*100	110	-100	

☑ Show Combined Values

Value Limits

From the Value Limits group box you can define the manner in which a variable value changes as a function of the SigGen Index (SGI) value. You can also define the range of allowable values for the variable. For combination variables, these values will only apply to primary variable(s), and will not limit the combination variable result. However, a zero will be used when the combination variable result becomes a divide-by-zero.



Value Limits

Default/Start 100

Step Size 100

Minimum 100

Maximum 20000

No. of Steps 200

Default/Start

Default/Start defines the value of the variable when the SigGen Index = 1. This is also the value for a constant variable. When prompt variables (Prompt Level 1, Prompt Level 2, Prompted File) and Dynamic variables are defined, this value is also the default variable value before the value is specified.

Step Size

Step Size must be defined when using the following functions:

- Linear step
- Log Step (base 2)
- Log Step (base 10)
- Randomize
- Random per SGI

Step determines how much the variable will be systematically varied as a function of the SigGen Index value. Default Step Size for a Randomized variable is 0, which causes SigGenRP to use the full floating point precision to generate the variable values. A zero Step Size for other functions will define a constant variable.

Minimum

In the Minimum field, you define the minimum allowable value for the variable.

Maximum

In the Maximum field, you define the maximum value allowed for the variable.

Note: Minimum value must be smaller than Maximum, and Default/Start must be within Minimum and Maximum range.

No. of Steps

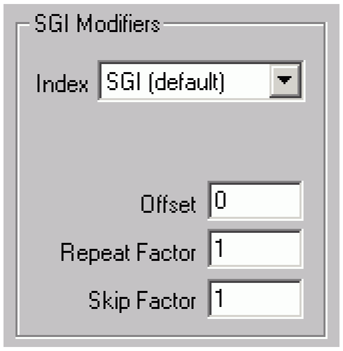
This field displays the total number of steps between the minimum and maximum variable values, when appropriate. This field is not editable.

SGI Modifiers

SigGen Index (SGI) is a unique number which starts at 1 and increments by 1 when a interval is presented. This SGI applies to every variable by means of SGI Modifiers. These SGI Modifiers become important when SigGen is used to design signals with complicated variables or nested variables.

By default, a stimulus presentation begins with the SGI = 1. Often, the stimulus signal generated at SGI = 1 is presented until a desired number of response signals have been attained. The SGI is then incremented by 1, then all variables and the entire signal are updated. This process continues until all SGIs have been presented. Through use of SGI Modifiers, it is possible to customize the process of SGI incrementation for individual variables. You may:

- Start variable calculation with an offset to SGI.
- Repeat SGI.
- Skip a specified number of SGIs.
- Create nested variables



SGI Modifiers

Index: SGI (default)

Offset: 0

Repeat Factor: 1

Skip Factor: 1

Index

By default, variables are indexed using the SigGen Index or SGI. SGI indexing can also be replaced by another variable with an optional scalar applied. This is useful for a multitude of applications when one or more variables should be calculated based on a single control variable. *(See Appendix D for more information on index variables.)*

Offset

The initial SGI is always 1. When SGI is used to calculate variables, any deviations from this default value are termed offsets. If you wish to start stimulus presentation with a variable calculated based on a modified SGI = *n*, then enter *n* in the Offset field. If the variable's values are 0, 1, 2, 3, 4, 5, ..., corresponding to SGI = 0, 1, 2, 3, 4, 5, ..., an Offset = 2 will result in modified SGI = 2, 3, 4, 5, 6, ..., and so the variable's values 2, 3, 4, 5, 6, ..., correspond to SGI = 0, 1, 2, 3, 4, 5, ...

The default value of Offset is 0, that is, the modified SGI equals to SGI. The value of Offset must be no less than -10,000 and no greater than 10,000.

Repeat Factor

The Repeat Factor field specifies the number of times, *r*, an SGI will be used as modified SGI. For example, when *r* = 2, modified SGI will be 1, 1, 2, 2, 3, 3, ..., corresponding to SGI = 1, 2, 3, 4, 5, 6, ...

By default, Repeat Factor is set to 1 (use SGI once). Repeat Factor may be no less than 1 and no greater than 100.

By using a Repeat Factor in tandem with a Termination Control specification of Loop (see next section), you can create nested variables.

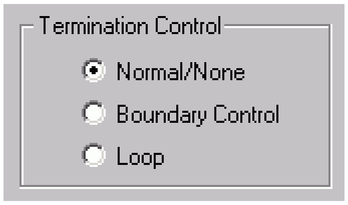
Skip Factor

The Skip Factor field specifies the number of the SGIs is to be ignored in the modified SGI. For example, if Skip Factor is 2, modified SGI will be 1, 3, 5, 7, 9, ..., corresponding to SGI= 1, 2, 3, 4, 5, 6, ...

By default, Skip Factor is set to 1. Skip factor may be no less than 1 and no greater than 100.

Termination Control

You can control the termination of SGI, so the signal play by choosing one of three types of termination control.



Termination Control
● Normal/None
○ Boundary Control
○ Loop

Normal/None

When Normal/None is chosen, the variable will not be considered in termination control. Thus, if a minimum or maximum variable value is reached, signal play will not terminate, nor will the variable value loop. Continuous signal generation will be based on the final variable value. In this case, signal play will be terminated by other variables or manually.

Boundary Control

Boundary Control provides a method of experiment termination. When Boundary Control is chosen, the SigGen Index (SGI) will not be incremented if such an action would cause the value of the variable to be:

- Less than the Minimum value
- Greater than the Maximum value

and signal play will halt. Such failure to increment the SGI produces differing results, depending on the application.

Boundary Control in SigGenRP In SigGenRP, when a boundary is reached and the SGI is no longer incremented, all signal generation halts.

Boundary Control in TDT's BioSigRP application When a boundary is reached and the SGI ceases to be incremented, signal generation will continue for the current SGI, but no more response data will be collected.

Note: Specification of Minimum and Maximum does not provide boundary control. Minimum and Maximum simply define the limits of parameter variation. The variable will not be allowed to go beyond either of these limits. With Boundary Control disabled, SigGen will continue to increment the SigGen Index and produce a signal during preview of multiple intervals, even if a Minimum or Maximum parameter value has been reached.

Loop

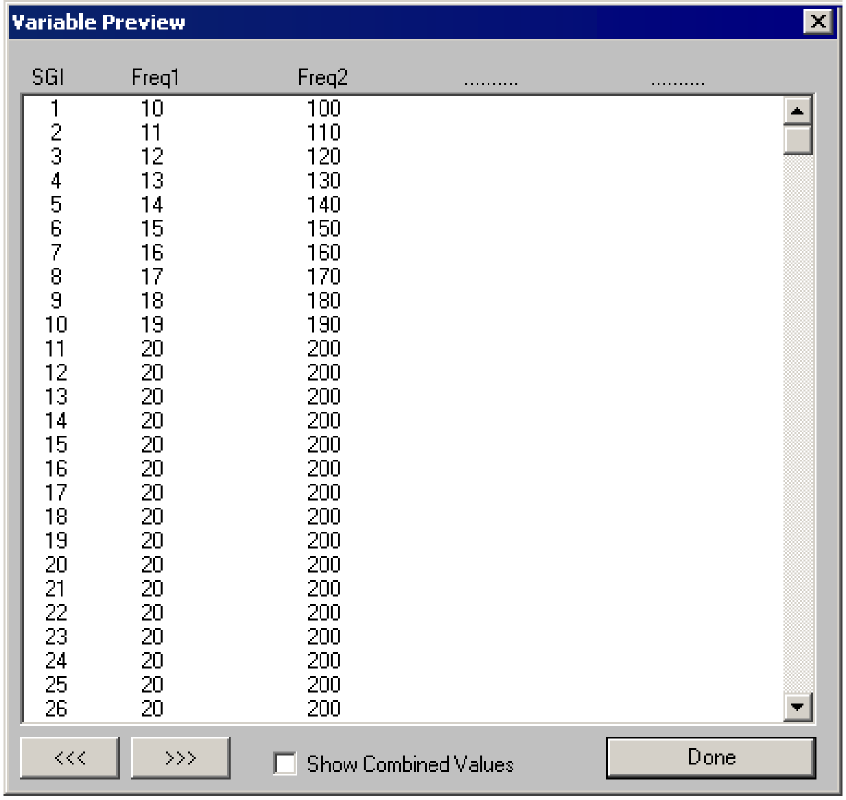
When Loop is chosen, parameters continuously loop through the series of variable values as SGI increments.

Previewing Variable Values

You may wish to preview the manner in which parameter values will vary as a function of the SGI. Variable values may be previewed from the Variable Preview dialog box.

To open the Variable Preview dialog box

- Click the Preview button.



SGI	Freq1	Freq2	....	....
1	10	100		
2	11	110		
3	12	120		
4	13	130		
5	14	140		
6	15	150		
7	16	160		
8	17	170		
9	18	180		
10	19	190		
11	20	200		
12	20	200		
13	20	200		
14	20	200		
15	20	200		
16	20	200		
17	20	200		
18	20	200		
19	20	200		
20	20	200		
21	20	200		
22	20	200		
23	20	200		
24	20	200		
25	20	200		
26	20	200		

The Variable Preview dialog box displays the current *stimulus schedule*. Variable values are shown for each SGI.

Scrolling

The Variable Preview dialog box can display up to five variables. You may view additional variables by scrolling to the right or the left.

To scroll to the right

- Click on the >>> button.

To scroll to the left

- Click on the <<< button.

To scroll up and down

- Use the vertical scroll bar.

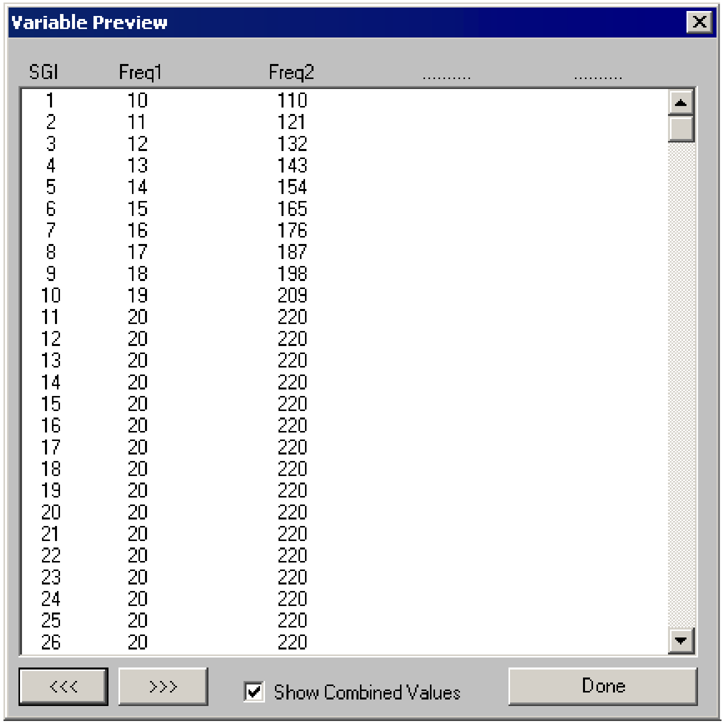
Showing Combined Values

You may have defined a combination variable. By enabling Show Combined Values, you may display the values of this combination variable as a function of the SGI. When Show Combined Values is not enabled, only their primary values will be displayed for combination variables.

To enable Show Combination Values

- Check the Show Combination Values box.

Enabling this feature would cause the previous stimulus schedule to appear as follows:



SGI	Freq1	Freq2	....	....
1	10	110		
2	11	121		
3	12	132		
4	13	143		
5	14	154		
6	15	165		
7	16	176		
8	17	187		
9	18	198		
10	19	209		
11	20	220		
12	20	220		
13	20	220		
14	20	220		
15	20	220		
16	20	220		
17	20	220		
18	20	220		
19	20	220		
20	20	220		
21	20	220		
22	20	220		
23	20	220		
24	20	220		
25	20	220		
26	20	220		

☑ Show Combined Values

Using Variable Files

When other functions cannot generate desired variable values, you may design custom variables using Variable Files. Variable files are also called Schedule files.

What Is a Variable File?

A variable (schedule) file is an ASCII text file that defines one or more user-customized variables by specifying the following:

- Variable Names
- Number of Repetitions for each variable

  Set this number equal to the number of times you want the signal repeated for a given variable value.
- A list of variable values for each variable

Creating a Variable File

Variable Files may be created with any editor that saves ASCII text files.

Default Naming Conventions By default, SigGenRP looks for files with the *.sch* file extension. It is a good idea to use this file extension when creating variable files.

Variable File Structure

Text Strings Most users will want to include some text information in their Variable File that describes the variable(s) defined. Such information can be placed at the top of the file, or between Variable Definition blocks.

Variable Definition One or more variables may be defined in a Variable File. Each variable is defined in a variable block. The beginning and end of a variable definition block are marked by a pair of brackets ({}). The variable definition block structure is as follows:

```
{
Variable Name
Number of signal repetitions, r
Variable value 1
Variable value 2
....
Variable value n
}
```

Below is a sample variable file.

```
Sample schedule file
It contains two variable definition blocks

This one controls my Delta_T, separation between masker and probe.
{
Delta_T
7
82.0
66.0
58.0
54.0
52.0
51.0
50.0
}

This is a list for intensity of the masker.  We'll run each intensity level
for all values of Delta_T.
{
Masker_Level
1
20.0
25.0
30.0
35.0
40.0
45.0
50.0
}
```

Nesting Variables

In some cases, you may wish to vary parameters in a nested fashion. For example, you may wish to vary the separation between a masker and its probe through the use of a variable such as Delta_T. For each Delta_T, you may in turn wish to vary the masker level.

The variable file shown above is an example of a two-level variable nesting. Please note the following:

- Variables are defined in the order of nesting.

  In this case, the separation between a masker and its probe, *Delta_T*, will be varied. For each value of Delta_T, the intensity of the masker, *Masker_Level*, will be varied. Thus, Delta_T is defined first; Masker_Level is defined second.

- The Number of Repetitions parameter is used to control repetitions during nesting.

  There are seven values for Masker_Level. This means that for each value of Delta_T, Masker_Level level will be varied seven times. This is indicated by setting the Number of Repetitions for Delta_T to seven. See line seven of the above variable file.

  For each Masker_Level, the signal will be played only once. This is indicated by setting the Number of Repetitions for Masker_Level to one. See line 21 of the above variable file.

Nesting variables can be also defined in SigGenRP without using a variable file. In the above cases, we can define Masker_Level with proper limits and Loop boundary control so it will loop over 7 values, and define Delta_T with a Repeat Factor of 7, we can obtain the same nested result.

Using a variable file you can define more complicated nesting variables, while you can only achieve a two-level nesting if you do not use a variable file.

Using Variables Defined in a Variable File

To access variables in a Variable File

1. Specify the file name.

   ➢ Type in the desired file name in the File Name field.

   or

   a. Click Find File.

   b. Select the desired file.

2. Type the variable name in the Name field.

Note: SigGenRP uses variable names to access variable values in variable files. The variable name typed in the Name field must match exactly the variable name assigned in the variable file.

Initializing Variables in a Variable File

When a variable is created using a variable file, variable values will be read as soon as the variable is accepted (clicking the OK button). However, when a file variable is edited (assigned with another variable file), or the variable file is altered, variable values may not be read automatically. At this time, when the current signal interval is previewed using Play button, the presented signal may not be the one with new variable values.

In many cases, the value of a file variable is read at the time it is assigned to a parameter. This value will be a function of the current SigGen Index (SGI). In some cases, however, the variable value is not automatically read upon assigning the variable to a parameter. This can cause problems if the parameter must fall within a specific range. It is a good idea to manually initialize all file variables prior to assigning them to parameters.

To manually initialize a file variable

➢ Click the Initialize button of the SigGenRP main window.

Using Variables to Control System 3 Parameter Tags

When the demands of the required stimulus are beyond SigGenRP's basic signal design abilities, SigGenRP can be used to control RPx device properties through parameter tags. Parameter tags are a powerful component of the RCO file and can be used to control various aspects of a stimulus such as programmable digital filtering, waveform generation, and even hardware such as programmable multiplexers (PM2r) and response boxes.

A good working knowledge of System 3 hardware and RPvdsEx is needed before modifying the SigGen RCO files.

System 3 Hardware and Software

For more information about creating RCO files, see the RPx help.

The RPx family of processors is an easy-to-use, easy-to-program data acquisition and stimulus presentation environment. Each RPx device contains analog outputs and inputs, digital inputs and outputs and a development program, RPvds. Compiled RPvdsEx files program the signal processor for stimulus generation and data acquisition using a drag-and-drop design. Circuits generated in RPvdsEx can be saved for use with other TDT software applications.

Chapter 4 Signal Segments - The Building Blocks of a Signal

When you are designing signals in SigGenRP, you are actually designing sub-signals, segments, and specifying how each segment is related to any others temporally. You are also specifying how a segment will affect the entire signal. This chapter introduces Segment design and its characteristics.

Segments, in turn, are composed of one to three components. From the Edit Signal Segments window, you can define segment components.

The Segment Window

The Edit Signal Segments window allows you to specify all the parameters necessary for the definition of a segment.

To access the Edit Signal Segment window

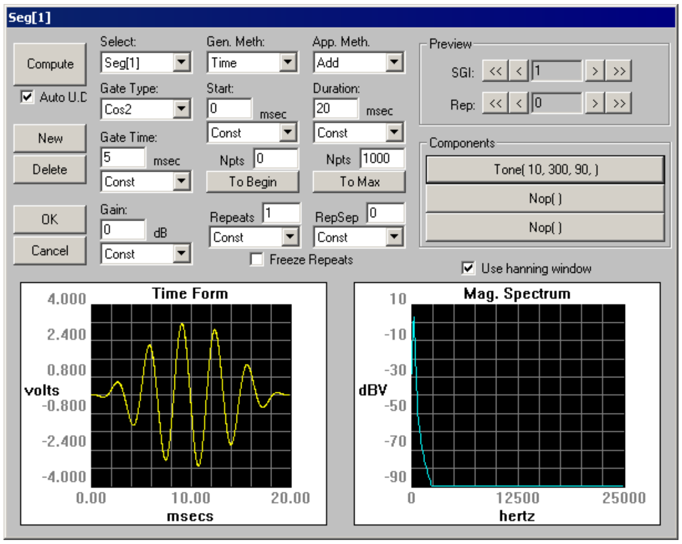
- Select Segment... from the Modify menu of the main window.

or

- Click the Modify Segment button on the toolbar.

or

- Double-click the desired segment marker.

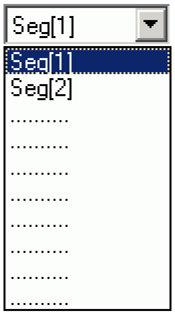


Seg[1]

Select: Seg[1]
Gen. Meth: Time
App. Meth. Add
☑ Auto U.D
Gate Type: Cos2
Start: 0 msec Const
Duration: 20 msec Const
Gate Time: 5 msec Const
Npts 0
Npts 1000
Gain: 0 dB Const
Repeats 1 Const
RepSep 0 Const
□ Freeze Repeats
Preview
SGI: 1
Rep: 0
Components
Tone(10, 300, 90,)
Nop( )
Nop( )
☑ Use hanning window

Creating/Selecting/Deleting a Segment

Select



Seg[1]

Seg[1]
Seg[2]
....
....
....
....
....
....
....
....

Existing segments are listed in the Select drop-down combo box.

To select an existing segment

- Click the desired segment in the Select box.

New

When you start to design a new signal, there is only one segment (empty) for the signal. You may need to create new segment(s) to complete the signal. New segments must be created in the Edit Signal Segment dialog box.

To create a new segment

- Click the New button.

Delete

A segment can be removed (deleted) from the signal.

To delete a segment

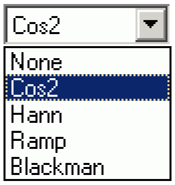
- Click the Delete button.

Note: A signal must contain at least one segment. You may not delete the last segment of a signal. If you wish to clear the entire signal, return to the SigGen main window and close the unwanted signal window**.**

Segment Generation Options

Windowing/Gating Parameters

Gate Type



From the Gate Type box, you can select a variety of gating or windowing functions, including:

Gating functions

- Cos^2 (10% - 90%)
- Ramp
- Blackman

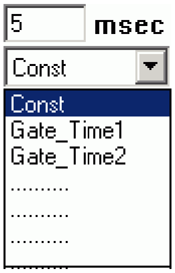
Windowing functions

- Hanning

In cases where gating or windowing is not necessary (e.g., click generation), choose:

- None

Gate Time



Gate Time specifies the rise/fall times of the gating function in milliseconds.

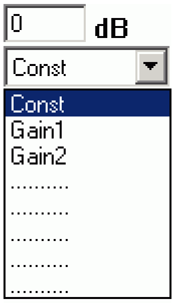
Constant or Variable Gate Time By default, Gate Time is defined as Const, and therefore remains constant throughout the preview process, regardless of the current SigGen Index value. Gate Time may be varied as a function of the current SigGen Index value by assigning a predefined variable (see Creating Variables).

To set Gate Time to a constant or a predefined variable

- ➢ Click Const or the desired variable name in the Gate Time box.
- ➢ If Gate Time is a constant, enter the value in the Gate Time field.

Signal Parameters

Gain



The Gain field of the Edit Signal Segments window is used to adjust the overall level of the digital segment generated by SigGenRP. All components of this segment are affected by this value.

Note: Gain is measured in dB. The reference level for dB is user defined (see Signal Setup).

Note: TDT's Real-time processors, as well as other equipment, limit the voltage range to ± 10 volts. Clipping will result during signal play if voltages exceed this range. If voltages of the segment, as displayed in the Time Form window of the Edit Signal Segments screen, exceed ± 10 volts, lower the value of Gain. You may have to lower the level of one or more of the segment's components to obtain a correct segment level. In contrast, when the voltage of the segment is too small, signals will not be correctly presented at the D/A converter even if the segment is smoothly displayed on the segment Time Form plot. If this segment is the dominant one in the signal, audible noise may appear. When this happens, you must increase the gain value to suppress noise, and use an external attenuator to adjust the signal.

Constant or Variable Gain By default, Gain is defined as Const, and therefore remains constant throughout the preview process, regardless of the current SigGen Index value. Gain may be varied as a function of the current SigGen Index value by assigning a predefined variable (see Creating Variables).

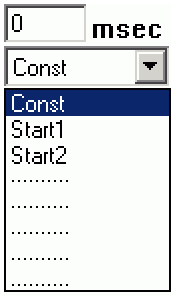
To set Gain to a constant or a predefined variable

➢ Click Const or the desired variable name in the Gain box.

If Gain is a constant,

➢ Enter the value in the Gain field.

Start



The value defined in the Start field defines the delay of the segment onset in the signal in milliseconds.

Constant or Variable Start By default, Start is defined as Const, and therefore remains constant throughout the preview process, regardless of the current SigGen Index value. Start may be varied as a function of the current SigGen Index value by assigning a predefined variable (see Creating Variables). You can arrange segments in various manners by assigning different (or variable) Start values for each segment.

To set Start to a constant or a predefined variable

➢ Click Const or the desired variable name in the Start box.

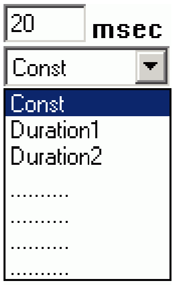
If Start is a constant,

➢ Enter the value in the Start field.

To set the Start value to the beginning of the signal

➢ Click the To Begin button found in the segment dialog box.

Duration



The value of the Duration field defines the total duration of the segment in milliseconds.

Constant or Variable Duration By default, Duration is defined as Const, and therefore remains constant throughout the preview process, regardless of the current SigGen Index value. Duration may be varied as a function of the current SigGen Index value by assigning a predefined variable (see Creating Variables).

To set Duration to a constant or a predefined variable

➢ Click Const or the desired variable name in the Duration box.

If Duration is a constant,

➢ Enter the value in the Duration field.

Note: Start and Duration values should not fall outside of the signal duration. If this happens and you do not correct it, SigGenRP issues a warning, truncates the segment, and continues.

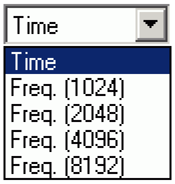
To set the Duration value to the entire length of the signal

➢ Click the To Max button found in the segment dialog box.

Generation Method

Gen. Meth (Generation Method)

You must specify how the segment (its components) is generated by SigGenRP when you design the signal. SigGenRP uses two basic generation methods: Time and Frequency. The Time method generates signal components by calculating their time waveform points. Frequency methods specify their frequency characteristics first, and use inverse FFT to obtain the waveform points. Depending on the frequency resolution, there are several Frequency method options.



To choose a Generation Method

- ➢ Click on the Gen. Meth. pull-down button and click on the desired Generation Method in the Generation Method combo box.

When to Use a Frequency Domain Generation Method

Generating signals in the frequency domain allows you to specify the frequency components of your signal. Some signals (tone, click, Gaussian noise, DC shift) can be generated from time domain or frequency domain, while others (band-limited noise, harmonics) can only be generated from frequency domain. Usually when you want to generate band-limited noise, harmonics, or to specify the frequency contents of a signal, you may consider using frequency method.

Choosing Among the Frequency Options

SigGenRP generates time-domain waveforms from frequency parameters through the use of the inverse Fast Fourier Transform (iFFT). The iFFT builds signals based on a finite number of frequency components (FFT points, which must be a radix-2 number). A list of frequency options is displayed in the Generation Method box. The number of combined magnitude and phase spectra FFT points is indicated in parentheses. To determine the number of either magnitude or phase spectra FFT points, simply divide this number by 2.

Choice of the number of FFT points depends on the *sample period* you have defined for the time-domain waveform and the desired *binwidth*, or distance between two adjacent point in the signal spectrum in Hz.

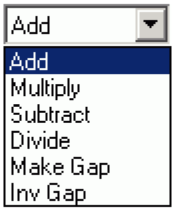
number of FFT points = 1/(binwidth · sample period)

To choose a frequency option

1. Calculate the desired number of FFT points using the above equation.
2. Choose the frequency option with a combined number of magnitude and phase FFT points greater than 2 times the number calculated in (1) in order to ensure a minimum binwidth. Since there are only a limited number of points for inverse FFT, the signal generated may not be enough to cover the duration of the segment. SigGen repeats the same iFFT result to obtain enough points.

Application Method

App. Meth (Application Method)



Segments may be applied to the existing signal in one of several ways:

- Add — A segment may be added to the existing signal.
- Multiply — A segment may multiply the existing signal.
- Subtract — Subtract a segment from the existing signal.
- Divide — Divide the existing signal by the segment.
- Make Gap — Creates a gap in the signal, based on the segment.
- Inv Gap — Creates an inverse gap in the signal, based on the segment.

Note: TDT's D/A converters, as well as other equipment, limit the voltage range to ± 10 volts. Clipping will result during signal play if voltages exceed this range. If this occurs, the overall signal, as displayed in the Signal Window of the SigGenRP main window, will appear clipped. When this occurs, some modification of segment level or component level will be necessary.

To choose an Application Method

➢ Click the desired method from the Application Method box.

Using Application Method to Build the Signal

Two Application Method options are designed to help build or modify a signal:

- Add
- Subtract

Add is used primarily to build a signal by adding together two or more segments. For example, a noise segment can be added to a signal to reflect the background noise.

Subtract is used primarily to remove unwanted signal components from the signal. For example, DC offset has to be specified in decibel values, so only positive DC offset can be produced in a segment. A negative DC offset can be generated by subtracting a DC offset from a signal.

Note: Segments consist of one or more components. Components are always additively applied to the segment. As part of component definition (see *Chapter 5*), it is necessary to define a component level. When using the Application Methods *Add* or *Subtract*, component level is defined in terms of the system calibration specified during signal parameter definition. For example, if system calibration is defined as 90 dB equal to 9 volts, specifying a 90 dB tone will result in a tone whose peak amplitude is 9 volts.

Using Application Method to Modulate the Signal

Several Application Method options allow you to modulate the current signal:

- Multiply
- Divide
- Make Gap
- Inv Gap

Note: Segments consist of one or more components. As part of component definition (see *Chapter 5*), it is necessary to define a component level. When using the modulating Application Methods presented above, component level is NOT defined in terms of the system calibration specified during signal parameter definition, but is instead defined as 0 dB equals 1 volt. Thus, specification of a tone with a level of 0 dB will create a tone with a peak amplitude of 1 volt.

The segment waveform generated will be displayed at the bottom of the Edit Segment Parameters dialog box in both Time Form and Magnitude Spectrum. The waveform can be previewed and updated in this box. Refer to Chapter 6, Working with Segments for detailed information.

Segment Repeating

To support the production of stimuli with multiple pulses, SigGenRP supports segment repeating. The exact number of repeats and each of the repeats' temporal position is controlled by three new parameters within the Segment edit dialog box.

The Repeats, RepSep, and Freeze Repeats options are used to control Segment repeating. When Repeats is set to zero or one, a single copy of the generated segment will be applied to the overall signal. Making the Repeats parameter two or more will cause the segment to repeat across the signal with a start-to-start separation of *RepSep* milliseconds between each segment onset.

Chapter 5 Components - The Building Blocks of Segments

Components are the foundation of a SigGen signal. Each component is a basic waveform of the signal. It is through the definition of component parameters that the time-varying waveform is defined.

How to Build a Segment - Signal Components

Components are the building blocks of a segment. The manner in which a component is generated is determined by the Generation Method assigned to the segment it is building.

Components may be generated in the following ways:

Generation Method	**Parameter specification**
Time	Time-domain parameters
Frequency	Frequency-domain parameters

Example: **Building a segment in the time-domain**

You are building a segment, seg1, composed of two component pure tones. You have chosen a Time as the Generation Method. Both of its components will be defined in the time domain.

How Components are Generated

SigGenRP generates a time-domain waveform for every component, regardless of which Generation Method, Time or Frequency, has been selected.

Time When time-domain parameters are specified, SigGenRP uses these parameters to generate a time-domain waveform. Specifically, SigGenRP utilizes built-in functions to calculate signal points when time method is used.

Frequency When frequency-domain parameters are specified, SigGenRP computes the magnitude spectrum of the signal based on those parameters, generates a phase spectrum, and then converts them into a complex buffer pair (real and imaginary buffer pair). An inverse Fast Fourier Transform (iFFT) will then be applied to the complex buffer pair to generate the time-domain waveform.

Combining Components

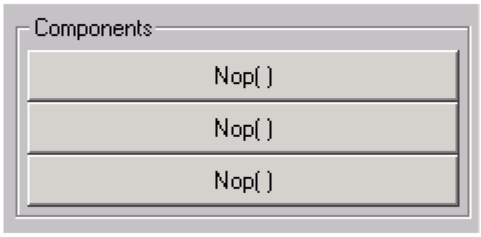
The time-domain component waveforms generated by SigGenRP are always combined additively.

Note: TDT's D/A converters, as well as other equipment, limit the voltage range to ± 10 volts. Clipping will result during signal play if voltages exceed this range. Combining components that are in phase may result in segment voltages in excess of ± 10 volts.

Designing Signal Components

Components may be designed from the Component Parameters dialog box. Component Parameter dialog box can be accessed from Edit Segment Parameter dialog box, Component group.

Components



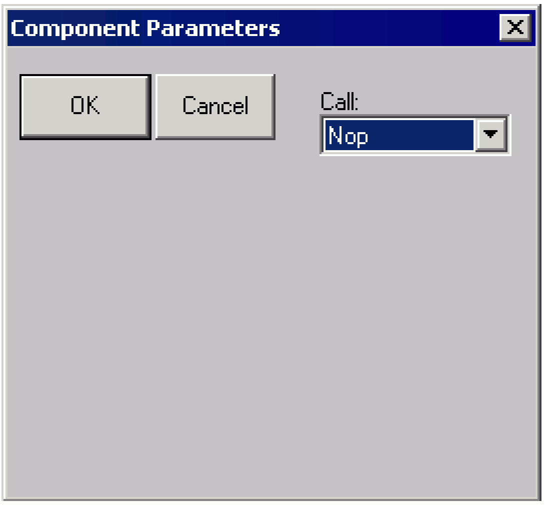
To open the Component Parameters dialog box

- Click the desired Component button.

  Component buttons read Nop[] (no operation) before a component has been added. After a component has been added, summary information is displayed on the button.

Component Parameters Dialog Box

Components of a signal segment are defined in the Component Parameters Dialog Box. Under Call there is a pull-down combo list box which contains all possible signal components SigGenRP can generate. Clicking on the pull-down arrow to display the list. There are different options for Time and Frequency methods.

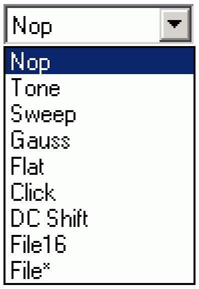


Call: Nop

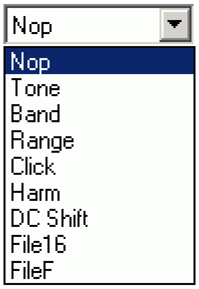
Call

The Call list box displays a list of options you may use in designing your segment. From this list you may choose the type of signal component you wish to define. Depending on which Generation Method you have chosen (Time or Frequency) you will see one of two lists of options:

Time

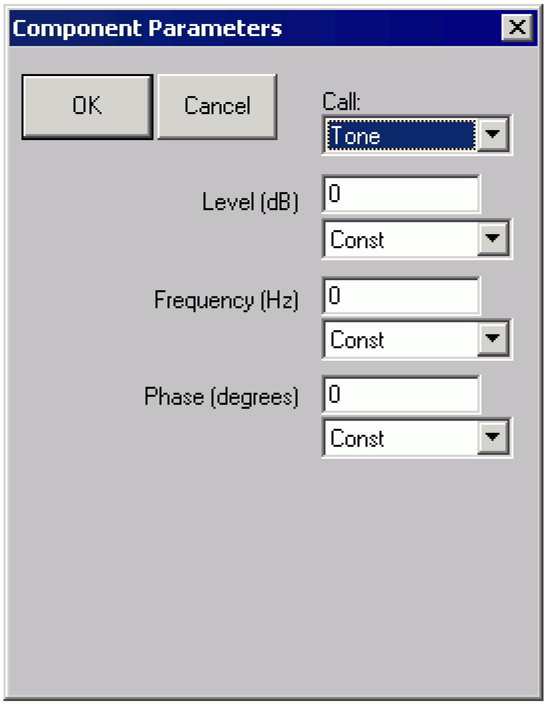


Frequency



Parameter Definition Fields

Once you have chosen the desired type of signal component, corresponding Parameter Definition Fields will be displayed in the dialog box. Each field specifies a parameter required for definition of the component. You must understand and fill the parameter values correctly for SigGenRP to generate the desired signal.



Component Parameters

Call: Tone

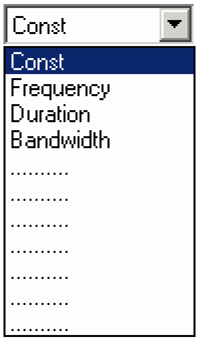
Level (dB) 0 Const

Frequency (Hz) 0 Const

Phase (degrees) 0 Const

Each Parameter Definition field is associated with:

- A Value field for entering a constant value
- A Variable box containing the names of all defined variables



Const

Const
Frequency
Duration
Bandwidth
....
....
....
....
....
....
....

To specify a constant value

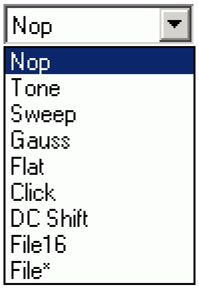
1. Choose Const from the Variables box (this selection is default).
2. Enter the desired value for the constant in the Value field.

To specify a variable value

➢ Choose the desired variable from the Variables box.

Note: In most cases, component definition requires the specification of a component level. When using the Application Methods, *Add* or *Multiply* (see *Chapter 4*), level is defined in terms of the system calibration specified during signal parameter definition. When using the modulating Applications Methods *Multiply*, *Divide*, *Gap*, *Inv Gap* (see *Chapter 4*), component levels of this segment are NOT defined in terms of the system calibration specified during signal parameter definition, but is instead defined as 0 dB equals 1 volt. For example, specification of a tone with a level of 0 dB will create a tone with a peak amplitude of 1 volt.

The Time Options



Nop Causes no operation to be performed, so no component will be defined. Choosing Nop for an existing component causes the component to be removed from the segment.

Tone Produces a pure tone waveform in the time domain. Parameters include:

- Level in dB
- Frequency in Hz
- Phase (cosine) in degrees

Note: The phase of the Tone component is considered as cosine degrees, that is, a tone with a zero degree phase will begin generating the waveform at the peak of a tone waveform, which is 90 degrees out of phase from a sine waveform.

Sweep Produces a frequency sweep (a tone with its frequency varying continuously). Parameters include:

- Level in dB
- Starting frequency in Hz
- Sweep rate in Hz/s
- Phase (cosine) in degrees

Sweep rate can be either positive or negative. When sweep rate is positive, the generated signal will have an increasing frequency, or when it is negative, a decreasing frequency. A pure tone will result if the sweep rate is specified to be zero.

Gauss Produces random noise with a Gaussian distribution. Parameters include:

- Level in dB

Note: The Gaussian distribution has a zero arithmetic mean, and in SigGenRP its standard deviation is the level specified above. If you generate a Gaussian noise with a specified level, the actual noise peak amplitude will be higher than the specified level.

Flat Produces random noise with a flat (uniform) distribution. Parameters include:

- Level in dB

Note: The waveform sample points of a flat noise are uniformly distributed between zero and the level specified above.

Click Produces a two-level (bi-polar) click. There are two peaks (positive and negative) in the bi-level click. Parameters include:

- Level one in dB
- Duration one in ms
- Level two in dB
- Duration two in ms
- Polarity (Enter +1 or -1, other values will affect the signal level.)

Two-level click can be considered as a pair of two opposite polarity clicks. Level one and Duration one specify the level and duration of the first click, and level two and Duration two, the second. When Polarity is a positive number, the first click will be a positive one, the second the negative one, and when Polarity is negative, the first will be negative, and the second positive. A zero Polarity will result in a Nop. The absolute Polarity value will also be used as a scaling factor to be multiplied to the generated component. For example, a Polarity value of 1.5 will result in a positive-negative click pair, and their peaks are 1.5 times of their levels.

If you want to generate a single-level click, you can specify a zero duration to either first or the second click. Durations must be either positive or zero. A negative duration will be taken as zero.

DC Shift Produces a DC offset (from 0 volt). Parameters include:

- Level in dB

Since Level of DC Shift operation is a dB value, only positive DC shift can be achieved with one DC shift component. To make a negative DC shift, you can generate a positive one first in one segment with desired level, and make another one in a new segment with twice the desired level and subtract it from the signal.

File16 This option allows you to define a component based on one or a group of headerless, 16-bit integer binary files. The file extension **must** be *.000*. When you use only one file, the real file extension must be 000. When you use a group of files, the file extensions must be three-digit numbers (for example, five possible file names: *file.005*, *file.004*, *file.003*, *file.002*, *file.001*, and only *file*.000 should be entered in the parameter field). Parameters include:

- File name (always *filename.000*)

- File extension variable. If there is only one file, Const should be selected. If more than one file will be used, a variable must be selected, and the variable values must be the three-digit numbers of the file extensions.

When a **File16** file is read to SigGen in time domain, the data points will be converted into voltage values based on 16-bit integer range of values (-32768 to 32767) and TDT D/A hardware range of output levels (±10V).

File* Allows you to define a component based on one or a group of sound files. The file extension **must** be *.000*. Refer to File16 for using more than one file. In addition to reading raw 16 bit integer and floating point data files, FILE* supports a number of file formats including the sound file format used on PCs, WAV files. In addition, FILE* includes arguments for controlling file seek position and the number of words read from the file, allowing FILE* to excerpt specified portions from files under variable control.

Finally, FILE* supports a versatile file-name access convention whereby the accessed file name is built using any combination of user defined text and a variable value. FILE*'s most powerful feature may be the indirect-file-access control whereby the named file contains a list of files names that are indexed using a specified variable to select the actual data file name.

Parameters include:

- File Type
- File Type variable
- Seek Position
- Seek Position variable
- Excerpt Size
- Excerpt Size variable
- Indirect File Offset
- Indirect File Offset variable
- File name (always *filename.000*)
- File extension variable. If there is only one file, Const should be selected. If more than one file will be used, a variable must be selected, and the variable values must be the three-digit numbers of the file extensions.

File Types By default, FILE* will load raw floating-point data without offset. When FILE* is selected, the first argument becomes the File Type to be accessed. This parameter can be specified with a variable, but care should be taken that the variable generates legal values that match the actual file type. It is possible to 'hang' the program by mismatching the file type specified with the actual file being accessed. The currently supported file types are:

Raw Float	index = 0
Raw Int16	index = 1
Ascii	index = 2
WAV (Mix)	index = 3
WAV (Left)	index = 4
WAV (Right)	index = 5

Raw Float Data samples are organized in a sequential floating point format using standard IEEE 32-bit (four byte) data. Numbers should be scaled between +/- 10.0 corresponding to the +/- 10 volt peak to peak output of the D/A converters.

Raw Int16 Data organized as a raw file of 16 bit integers stored in MSB first format. This component call is compatible with the INT16 file format supported by SigGenRP and includes the other features of FILE* like indexing etc. Data should be scaled between +32767 and –32768 which will be converted to +/- 10volts when loaded.

Ascii This is a standard text list of floating point numbers. These lists can be generated with EXCEL or MATLAB. Data should be scaled between +/- 10.0 as with Raw Float. It should be noted Ascii data files may be quite large with a single data value taking over 20 bytes of disk space.

WAV Files SigGenRP supports the Windows WAV file format. While WAV files can be either 8- or 16-bit, SigGenRP only supports 16-bit files and will fail to load if an 8-bit file is selected. A number of free-ware Windows programs are available for editing and formatting WAV files. These programs typically allow the user to get the file into the proper 16 bit format with the needed sample rate.

NOTE: the FILE* loader will NOT convert the sampling rate of the loaded file to match that specified within SigGenRP. It is the users responsibility to ensure the WAV files being used have the appropriate sample rate.

WAV (mix), WAV (left) and WAV (right) Mono and Stereo WAV files can be read. When mono files are used, any of the WAV component calls will load the same data. If stereo files are used, use the mix, left and right options to load the needed channel. WAV (mix) sums the left and right data and scales down by 6dB.

Seek Position

Seek position and excerpt size are used to select the portion of the sound file you wish to load. The SEEKPOS parameter can range from 0 to files_size – points-to-read. Seeking beyond the end of the file will cause garbage to load. It is the user's responsibility to limit file access to the files size. The SEEK_POS parameter is useful for accessing random segments from a pre-computed noise file. Often masking noise is needed with specific frequency characteristics. A large noise file can be pre-computed and saved to disk. The SEEK_POS argument can then be fed with a RANDOMIZED variable accessing random noise segments generating a pseudo-random noise effect.

Excerpt Size

The EXCERPT SIZE parameter is used to select the number of points to be read from the file. Normally (EXCERPT SIZE = 0) FILE* will read enough points to fill the target segment. That is a 100 ms segment sampled at 50KHz will load 5000 points. The EXCEPRT SIZE parameter can be used to override this behavior, by limiting the number of points read to something less than the 5000 needed to fill the segment. This argument is very useful for reading sound segments from raw data files with variable lengths. Also if one wishes to take excerpts from a larger file, this parameter is useful for controlling the number of points read into the segment.

File Name

FILE* supports a versatile file naming/lookup scheme, allowing the output of a variable to be inserted at any point within a text field specifying the file name. The examples below illustrate how the file name parameter functions.

Simple fixed file access.

File Name	=	"sound.wav"		
SigGen Variable	=	Const		
File Accessed	=	'sound.wav'		

*** Wildcard file access Example 1.** Will access file with the name built using the text specified and the current value of the variable named QSPEC.

File Name	=	"snd***.wav"		
SG Variable	=	QSPEC	=	247.0
File Accessed	=	'snd247.wav'		

*** Wildcard file access Example 2**. Builds file name as specified and truncates the number to the integer portion.

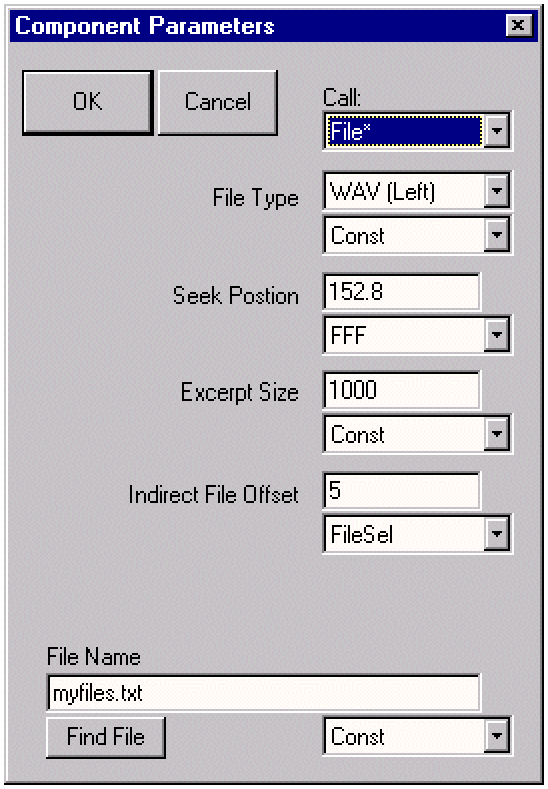
File Name	=	"sound.g**"		
SG Variable	=	QSPEC	=	12.8
File Accessed	=	'sound.g12'		

*** Wildcard file access Example 3.** Builds file name using first three digits of current value of QSPEC.

File Name	=	“snd***zz.i16”		
SG Variable	=	QSPEC	=	5437.0
File Accessed	=	‘snd543.i16’		

Indirect File Offset

When this parameter is made non-zero, indirect file accessing is enabled. This feature supports indirect file accessing whereby the File Name parameter (described above) accesses a text file containing a list of sound file names each on its own text line. Each line of the file must have the exact same number of characters and the first line is not used (index=0). The value of the Indirect File Offset parameter is used to index to a specified line and get the actual sound file name. Consider the following example.



Call: File*

File Type: WAV (Left), Const

Seek Postion: 152.8, FFF

Excerpt Size: 1000, Const

Indirect File Offset: 5, FileSel

File Name: myfiles.txt, Const

Myfiles.txt contains the following text:

```
1234567.123
f1k_cat.wav
f1k_dog.wav
f1k_mou.wav
f1k_frg.wav
```

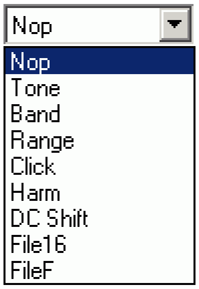
```
f1k_bat.wav
f1k_zea.wav
```

The WAV file accessed will be f1k_bat.wav. 1000 points will be read starting at point 152 and if the file is stereo, left channel data will be returned. Note that the first line is not used, and all lines have the same number of characters.

File16 and **File*** are very useful ways to generate waveform components which cannot be easily produced with simple mathematical functions. Combined with SigGenRP variables, you can assign several binary files to the same component, and use a certain file for a certain interval based on the variable values.

When a **File*** file is read to SigGenRP in time domain, the values of the data points will be taken in terms of volts and no conversion will be performed.

The Frequency Options



Note: Components generated from Frequency Method are periodic waveforms. Each "period" of the waveform includes the same number of points of inverse FFT (specified as Freq[####] in Edit Segment Parameters dialog box). One "period" of the generated tone has an integer number of complete cycles. So, generated noises may appear periodic due to this inverse FFT technique. You may want to select a different number of iFFT points to produce a satisfactory signal.

Nop Causes no operation to be performed, so no component will be defined. Choosing Nop for an existing component causes the component to be removed from the segment.

Tone Produces a pure tone waveform from frequency domain. Parameters include:

- Level in dB
- Frequency in Hz
- Phase (cosine) in radians

Band Produces a noise of a specific bandwidth around a center frequency. Parameters include:

- Low frequency level in dB: The lower end level of the band. This level is per Hz average level.
- High frequency level in dB: The higher end level of the band. This level is per Hz average level.

- Center frequency in Hz: The geometric center frequency. The high and low frequencies are calculated based on following:

$$F_H = F_C \times 2^{BW/2}$$

$$F_L = F_C \times 2^{-BW/2}$$

Where: F_H is the high frequency. F_L the low frequency, and F_C the center frequency, and BW the bandwidth in Octaves.

- Bandwidth in Octaves: One Octave is a doubling (or halving) usually in frequency measurements. For example, one Octave from 200Hz is 400Hz, while two Octaves is 800Hz, and so forth. This value must be greater than zero. A zero will produce a tone, and a negative value will generate an error message.
- Roll-off in dB/Octave: The roll-offs at the band boundaries. A negative value specifies a roll-off, while a zero or positive value sets the roll-off to be infinite.

Range Produces a noise of a specific frequency range. Parameters include:

- Low frequency level in dB: Magnitude spectral level at the lower end of the range.
- High frequency level in dB: Magnitude spectral level at the higher end of the range. Levels at other frequencies will be linearly interpolated.
- Low frequency in Hz
- High frequency in Hz
- Roll-off in dB/Octave (see Band)

Click Produces a bi-level click from frequency domain. Parameters include:

- Low frequency level in dB
- High frequency level in dB
- Low frequency in Hz
- High frequency in Hz
- Roll-off in dB/Octave

Harm Produces a spectrum composed of a fundamental frequency and several of its harmonics. Parameters include:

- Fundamental frequency level in dB
- Fundamental frequency in Hz
- Frequency separation in Hz: Distance between the frequency components.
- Roll-off in dB/component: The magnitude reduction rate in dB per component. A positive number indicates an increase in magnitude.
- Number of components: Unity means only the fundamental component (tone), zero indicates infinite number of harmonic components. A negative number is invalid, and may crash your system.

DC Shift Produces a DC offset in frequency domain. Parameters include:

- Level in dB: Since a DC-offset in spectrum is the same level of DC offset in time signal, this level is both time and frequency domain DC offset level. (See *The Time Options* if you want to generate a negative DC Shift)

File 16 and FileF

These options allow you to define a component based on one or a group of headerless, 16-bit integer binary files or binary floating point files. When you use a group of files, the file extensions must be three-digit numbers (for example, five possible file names: *file.005, file.004, file.003, file.002, file.001,* and only *file*.000 should be entered in the parameter field). Parameters include:

- File name (*filename.000 for more than one file*)
- File extension variable. If there is only one file, Const should be selected. If more than one file will be used, a variable must be selected, and the variable values must be the three-digit numbers of the file extensions.

The frequency and phase data should be **interleaved** (freq, phase, freq, phase …). The level of each frequency bin is specified in dBVolts and the phase in degrees (0 to 360).

The values of the frequency bins should be set to something like -250 if you don't want sound, and 0 if you want sound (for example, a 0 in one bin would produce a 1 V tone). It is important that you have the same number of frequency points and phase points as you have points in your FFT. For example, if you use a 1024-pt FFT, your file should contain 2048 points (1024 freq points, and 1024 phase points). If it is too short, SigGen will include a bunch of zeroes at the end, which results in a large unwanted signal. If you have too many points, SigGen will use the number of points of the specified FFT, and ignore the remaining values. Note, this will ignore the calibration you set in SigGen (e.g. 9 V = 180dB).

Note: Entering 0 for Roll-off in dB/Octave will result in the computation of infinite roll-off.

Note: In both time domain and frequency domain, data points in **File16** and **FileF** files must have the same sampling period as the signal you are designing.

Importing Components

You may desire to create a SigGenRP signal based on one or several existing non-SigGenRP digital signals. SigGenRP signals are built from segments. Segments are built from components. A non-SigGenRP signal, therefore, must be imported as a component.

Imported waveforms may be specified in either the time or frequency domains. Refer to the above sections for designating file extensions.

Importing Time-domain Waveforms

Time-domain waveforms may be imported for a variety of reasons.

- Including the waveform as one segment of a multi-segment signal.
- Adding the waveform to another component or segment.
- Multiplying another segment by the imported waveform.

To import a time-domain component

1. If you want to import more than one file, define a variable which will have values equal to the file extension digits while SGI is increasing.
2. Access the Edit Signal Segments window.
3. Choose Time from the Generation Method list box.
4. Click one of the three Component buttons to access the Components Parameters dialog box.
5. Choose the desired File command from the Call list box.

   To import a 16-bit integer binary file,

   ➢ Choose File16.

   To import another file type,

   ➢ Choose File*.
6. Type in the file name in the File field. The file extension must be *.000*.
7. If you want to import more than one file, select the file extension variable in the variable list.
8. Click the OK button.

Importing Frequency-domain Waveforms

You may wish to import a component defined in the frequency domain. An imported frequency domain waveform may be used to:

- Generate a time-domain component waveform with spectral characteristics that remain constant over time.
- Apply spectral shaping to other SigGenRP components.

To import a frequency-domain component

1. If you want to import more than one file, define a variable which will have values equal to the file extension digits while SGI is increasing.
2. Access the Edit Signal Segments window.
3. Choose Time from the Generation Method list box.
4. Click one of the three Component buttons to access the Components Parameters dialog box.
5. Choose the desired File command from the Call list box.

   To import a 16-bit integer binary file,

   ➢ Choose File16.

   To import a floating point binary file

   ➢ Choose FileF.
6. Type in the file name in the File field. The file extension must be *.000*.
7. If you want to import more than one file, select the file extension variable in the variable list.
8. Click the OK button.

Chapter 6 Working with Segments

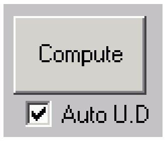
After you have defined component parameters and created a segment, you may:

- Modify segment parameters.
- Obtain information about the segment's time-domain waveform.
- Obtain information about the segment’s the frequency spectrum.

You may do these things from the Edit Signal Segments window.

Computing Segments

As you modify segment parameters, you affect the characteristics of the segment waveform. SigGen does not automatically regenerate the waveform every time you change a parameter. A recalculation and updating will be performed when you click OK to finish editing segments. It may be necessary to recalculate the segment while you modify the parameters. This can be done at various points in the modification process.



Manual Update

You may elect to re-compute segments manually. This allows you to edit segment parameters without recalculating the segment.

Compute

Clicking the Compute button recalculates the segment based on the currently specified parameters.

Auto Update

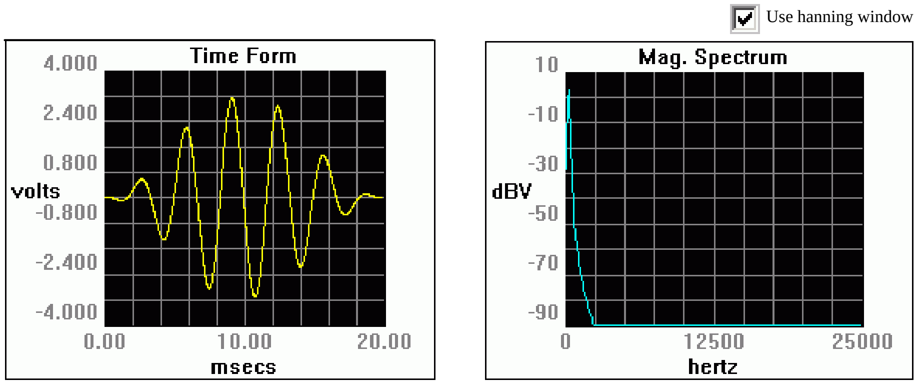
You may wish to have SigGenRP automatically recalculate segments when a parameter value is changed.

Auto U.D.

When Auto Update is checked, the segment is automatically recalculated whenever you move the focus from a field that has been updated. The following are examples of situations where recalculation would occur.

- You edit a parameter and then TAB to another field.
- You edit a parameter and then click in another field.
- You select a new parameter from a list.

Displaying Segments


☑ Use hanning window

After you have created a segment, SigGenRP displays the time-domain waveform in the Time Form window. The magnitude spectrum of the segment is displayed in the Magnitude Spectrum window.

The Time Form Window

The Time Form window displays the time-varying waveform of the current segment. The x-axis is measured in milliseconds. The y-axis is measured in volts.

Default Display

x-axis   By default, values of the x-axis extend from 0 to the number of milliseconds defined in the Duration field of the Edit Signal Segments window. You may use cursor to zoom-in to display details.

y-axis   SigGen automatically scales the y-axis so that the peak voltage of the signal is displayed. The x-axis is always centered at the y-axis (0 volt horizontal line). y-axis is always placed at the left boundary of the time plot.

The Magnitude Spectrum Window

The Magnitude Spectrum window displays the frequency domain values of the segment. The x-axis is measured in Hertz. The y-axis is measured in dBV (decibels in Volts). The reference value for dBV is user defined (see Calibration).

Default Display

x-axis The x-axis extends from 0 Hz to the Nyquist frequency based on the current sample period.

Nyquist frequency (Hz) = 1/sample period (seconds) · 1/2

y-axis SigGenRP automatically scales the y-axis so that the maximum value in dBV can be displayed clearly.

Using a Hanning Window

You may apply a Hanning window to the frequency domain signal by checking the Use Hanning Window box.

Using Cursors

You may wish to focus on a portion of either the Time Form window or the Magnitude Spectrum window. SigGenRP provides two types of cursor functions that you can use to zero-in on specific areas of the display.

Zoom-In

You can zoom-in on a section of either the Time Form window or the Magnitude Spectrum window.

To zoom in

1. Place the mouse at the right or left edge of the desired area of either the Time Form window or the Magnitude Spectrum window.
2. Hold down the left mouse button.
3. Drag the mouse over the desired area.
4. Release the left mouse button.

   The window will now display the desired section.

Note: When you zoom-in on a Time Form window, the Magnitude Spectrum window is automatically recalculated to reflect the new spectrum of the zoomed-in section.

To return to the default display

➢ Click the right mouse button anywhere in the display window.

Crawling

You can "crawl" through a zoomed-in time-domain waveform. Crawling will scroll the display through the waveform.

To crawl to the right

- Click the left mouse button in the right portion of the display window.

To crawl to the left

- Click the left mouse button in the left portion of the display window.

Note: The magnitude spectrum will be automatically recalculated when the time domain window is scrolled in order to reflect the energy represented in that portion of the signal that is displayed in the window.

Placing a Cursor

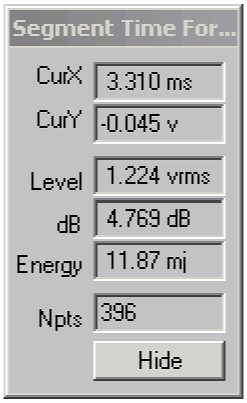
You can obtain information about a specific point in either the Time Form window or the Magnitude Spectrum window by placing a cursor within the desired display. The information will be displayed in the Segment Time Form, described on the next page.

To place a cursor

1. Place the mouse pointer in the desired display window.
2. Hold down the Shift key.
3. Hold down the left mouse button.
4. Drag the cursor to the desired location.
5. Release the Shift key and left mouse button.

The Segment Time Form

The Segment Time Form displays information about the current segment or portion of a segment. The Segment Time Form may display information about the Time Form window or the Magnitude Spectrum Window.



Segment Time For...

CurX 3.310 ms

CurY -0.045 v

Level 1.224 vrms

dB 4.769 dB

Energy 11.87 mj

Npts 396

CurX

CurX is the x-axis (time) value at the cursor, in milliseconds (ms).

CurY

CurY is the current y-axis value of the signal at the point of any cursor, in volts.

Level

Level is the root mean squared (RMS) value of the signal as measured in volts.

dB

dB is the level in dB Volts of the currently displayed signal.

Energy

Energy is the total energy of the displayed signal, in millijoules (mJ).

Npts

Npts is the number of sampling points contained in the displayed time signal waveform.

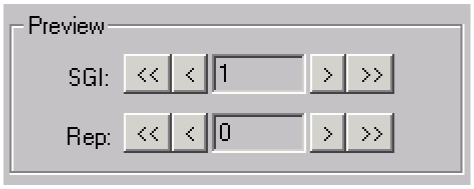
Hide

Clicking the Hide button causes the Segment Time Form to be hidden. You may display it again by selecting Cursor Box from the View menu of the main window.

Previewing Segment Intervals

Individual segment can be previewed in the Time Form and Magnitude Spectrum windows of the Edit Signal Segments window. Using SGI you can preview the segment in any interval of the signal.

Specifying the SigGen Index



SGI: 1

Rep: 0

SigGen Index

The SigGen Index (SGI) field specifies the specific interval you wish to preview. Since variable SGI is defined in Edit Signal Parameters window, boundary control is not applied to the SigGen Index here. In other words, you can increment SGI beyond its actual value. In this case, the last variable values will be used to compute the waveform.

<	>	Lowers/raises the trigger tick in increments of 1.
<<	>>	Lowers/raises the trigger tick in increments of 10.

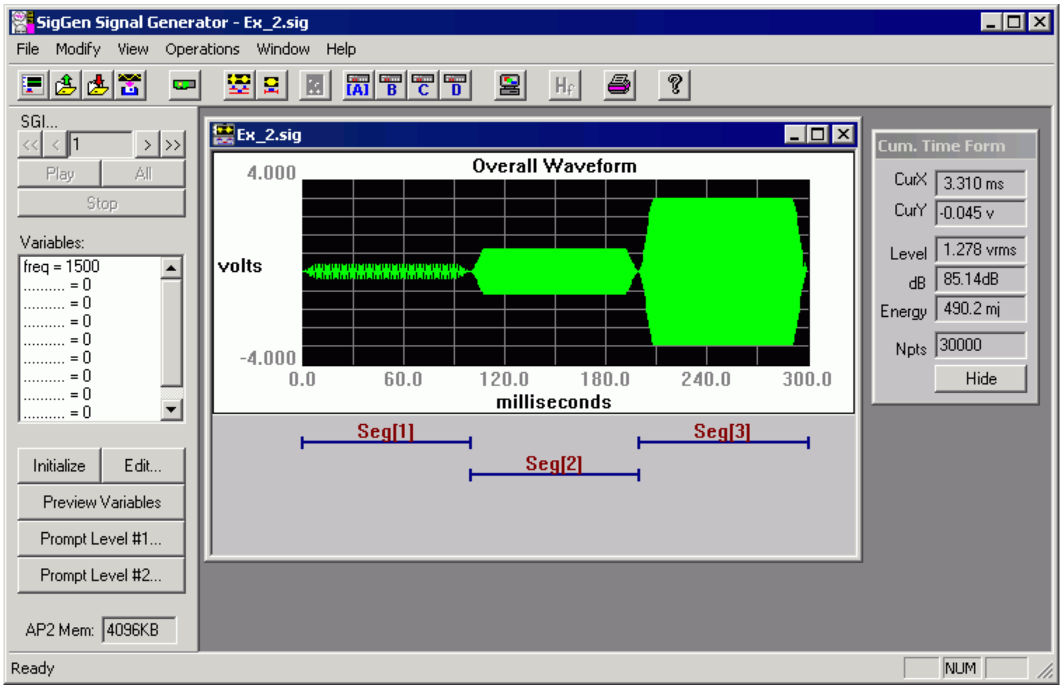
Note: The Rep. Field is not used at this time.

Chapter 7 Working with the Complete Signal

Once you have defined all signal-level, segment-level, and component-level parameters, you can view the final waveform and preview the signal.

To view or preview the overall signal

➢ Return to the main SigGen window.



SGI... 1

Variables:

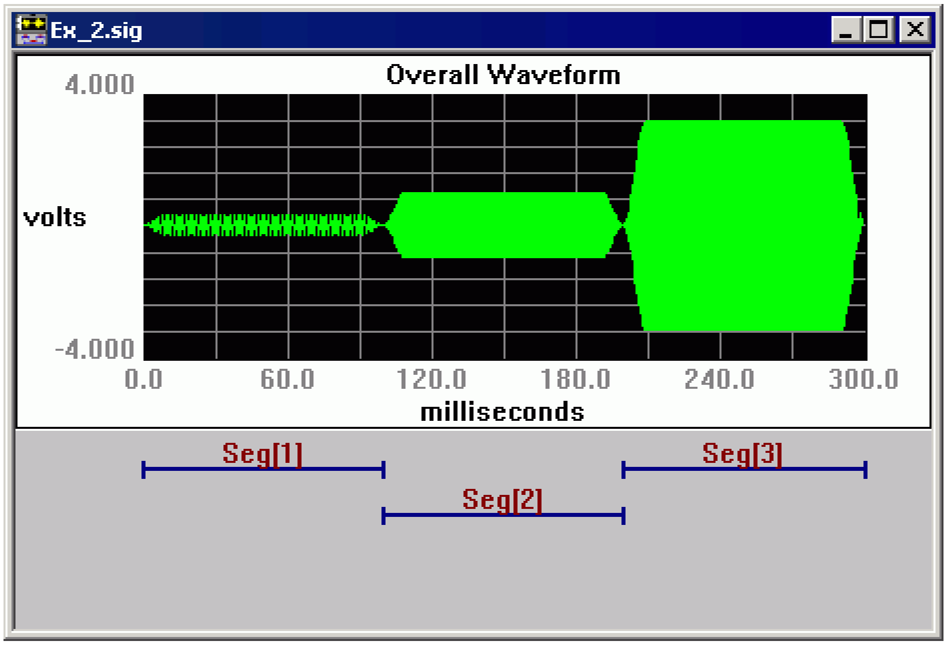
freq = 1500
.... = 0
.... = 0
.... = 0
.... = 0
.... = 0
.... = 0
.... = 0
.... = 0

AP2 Mem: 4096KB

CurX	3.310 ms
CurY	-0.045 v
Level	1.278 vrms
dB	85.14dB
Energy	490.2 mj
Npts	30000

Displaying the Signal

The overall signal is now displayed in the current Signal window.



When you click OK to return to the current Signal window from either Signal Parameters or Edit Signal Segments window, SigGenRP will automatically update all parameters and the overall signal waveform. The Signal window displays the time-varying waveform of the current segment and the x-axis is measured in milliseconds. The y-axis is measured in volts. If you have defined/modified a variable file, you may click on the Initialize button to read in all variable values and recalculate the waveform.

Default Display

x-axis By default, values of the x-axis extend from 0 to the number of milliseconds defined in the Duration field of the Signal Parameters window.

y-axis SigGenRP automatically scales the y-axis so that the peak voltage of the signal is displayed. The origin of the y-axis is always 0 volts.

Note: TDT's D/A converters, as well as other equipment, limit the voltage range to ± 10 volts. The Signal window limits the displayed voltage to this range. If there are points of which the calculated values are greater than 10 volts, or smaller than -10 volts, you will spot clipping from this display. When this happens, if you are concerned, you may change either the segment levels or component levels.

Using Cursors

It may be desirable to look at the details of a small signal portion, or obtain information on specific point. You can zoom-in on a portion of the overall signal or place a cursor on a specific point in the overall signal.

Zoom-In

To zoom in

1. Place the mouse at the right or left edge of the desired area of the Signal window.
2. Hold down the left mouse button.
3. Drag the mouse over the desired area.
4. Release the left mouse button.

   The window will now display the desired section.

To return to the default display

➢ Click the right mouse button anywhere in the Signal window.

Crawling

You can "crawl" through a zoomed-in time-domain waveform. Crawling will scroll the display through the waveform.

To crawl to the right

➢ Click the left mouse button in the right portion of the display window.

To crawl to the left

➢ Click the left mouse button in the left portion of the display window.

Placing a Cursor

You can obtain information about a specific point in the Signal window by placing a cursor with the desired display. This information is displayed in the Cumulative Time Form, as described on the next page.

To place a cursor

1. Place the mouse in the desired display window.
2. Hold down the Shift key.
3. Hold down the left mouse button.
4. Drag the cursor to the desired location.
5. Release the Shift key and left mouse button.

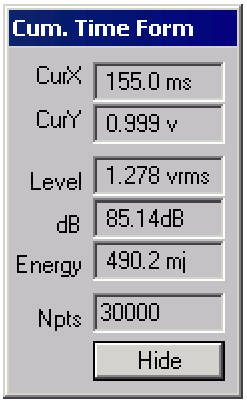
The Cumulative Time Form

The Cumulative Time Form (or Cursor Box) displays information about the current signal or portion of a signal. It is displayed by default when SigGenRP opens, but may be hidden at any time.

When there is no cursor in the signal waveform, CurX and CurY in the Cumulative Time Form will be empty.

To display the Cumulative Time Form

➢ Select Cursor Box from the View menu.



CurX 155.0 ms

CurY 0.999 v

Level 1.278 vrms

dB 85.14dB

Energy 490.2 mj

Npts 30000

CurX

SigGenRP allows you to place a cursor within the displayed signal. CurX is the x-axis (time) value of the cursor, in milliseconds (ms)

CurY

CurY is the current y-axis value at the cursor, in volts.

Level

Level is the root mean squared (RMS) value of the displayed signal measured in volts.

dB

dB is the average level, in dB Volts, of the currently displayed signal

Energy

Energy is the total energy of the displayed signal, in millijoules (mJ).

Npts

Npts is the number of sampling points contained in the displayed signal waveform.

Hide

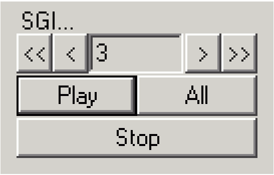
Clicking the Hide button causes the Segment Time Form to be hidden. You may display it again by selecting Cursor Box from the View menu of the main window. You can also hide the Cumulative Time Form by clicking Cursor Box on the View menu.

Previewing the Signal

While SigGen can design signals for a variety of purposes, many users will wish to design stimulus signals for experiments. Often, these experiments will require the repeated presentation of the stimulus signal through multiple intervals. Signal parameters may be varied from one interval to another during the experiment. SigGen provides a means for previewing (listening to) the signal through all intervals, or SigGen Indexes (SGIs).

The SigGen Index

By setting the SigGen Index (SGI), you can play one selected interval of your signal or play all intervals of your signal.



SigGen Index

You can preview a specific interval, *n*, of the signal by setting the value of the SigGen Index equal to *n*. This is most useful if at least one signal parameter will vary from one interval to another. Boundary control applies to this SGI display, so if your variables are limited by a boundary, you can only change SGI in the range to view the signal.

<	>	Lower/raise the SigGen Index in increments of 1.
<<	>>	Lower/raise the SigGen Index in increments of 10.

Previewing

Play

Clicking the Play button plays interval *n* of the signal where *n* is the number specified in the SigGen Index field (the currently displayed signal).

All

Clicking the All button plays all specified intervals of the displayed signal. When All is clicked, SigGenRP will initialize all variables, recalculate the entire waveform, and present every signal interval as SGI increments. A next interval is calculated when the previous interval is played, so the signal intervals will not be presented continuously. The delay between intervals is determined by the time SigGen takes to calculate the next interval.

Stop

Clicking the Stop button stops the signal preview (either the Play or play All mode).

Note: When SigGen is started and it cannot find a D/A device (the hardware may not be turned on), it will still run but the above three buttons will be dimmed and not be available for signal preview.

Part

2

Illustrative Examples

Chapter 8 General Examples

The examples presented in this chapter illustrate the use of a variety of SigGen features. Each example introduces new SigGen concepts. The examples found in this chapter are listed below. Also listed are the concepts introduced in each example.

Example 1: Generating a Tone Blip

- Signal
- Segments (SPL)
- Components
- Sampling Period
- Duration
- Calibration in dB
- Tone
- SigGen Files

Example 2: Generating a Three-Tone Blip Series

- Variable Intervals
- SigGen Index
- SGI Modifiers
- Previewing Signal
- SigGen Wave File
- Gating

Example 3: Generating an Amplitude-Modulated (AM) Signal

- Application Methods
- AM Signal

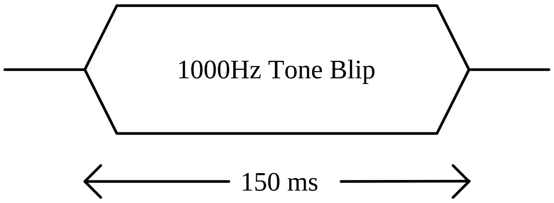
Example 4: Generating a Multi-Component Signal

- Variable file
- Using a Disk File (FileF)

Example 5: Generating a Band-Limited Gaussian Noise

- Frequency Method
- Segment Level
- Variable File
- Normalization File

Example 1: Generating a Tone Blip



This example will show you how to generate a single-tone blip signal with a frequency of 1 kHz, a level of 80 dB, a total duration of 150 msec, and a sampling frequency of 50 kHz (sampling period = 25 µsec). The calibration reference will be 80 dB = 1.0 V peak voltage. Finally, this signal will be previewed and then saved to a *.sig* file.

Concepts introduced:

- ✓ *Signal*
- ✓ *Segments*
- ✓ *Components*
- ✓ *Sampling Period*
- ✓ *Duration*
- ✓ *Calibration*
- ✓ *Tone*
- ✓ *SigGen Files*

Throughout this example, several SigGen concepts will be discussed. These concepts are listed in the adjacent column, and will be marked by the SigGen icon when they are introduced.

To generate, preview, and save the single-tone blip signal, you will need to perform the following steps:

1. Run SigGen.
2. Define the signal parameters.
3. Create the segments and components.
4. Preview the signal.
5. Save the signal.

Note: Only those parameters to be entered or selected are given below. All other parameters will remain according to the default settings.

Run SigGen and Create a New Signal

To run SigGen

➢ Double-click the SigGen icon

or

➢ Select SigGen from the Start\Program menu.

To create a new SigGen signal

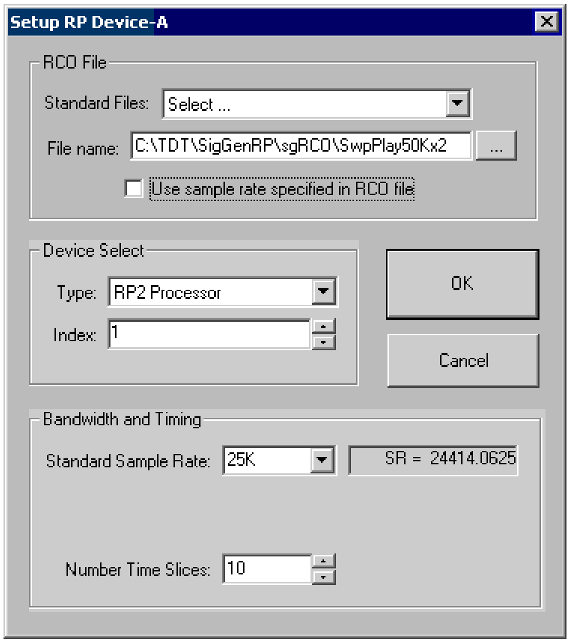
➢ Select New from File menu, or click on the New Signal icon on the toolbar.

Configure the RPx device and RCO file

For System 3 equipment SigGenRP uses the RPx device and RCO files (See RPvdsEx help) to play out signals. The

To configure the RPx device and select the RCO file

1. Select RP devices|Device A from the Modify menu or click on the “A” icon on the toolbar.
2. Select from the Standard file menu “Sweep Play, 2 Chan 50K (max 100K)”.
3. Select RP2 Processor as device type
4. Select Index 1.
5. Deselect “Use sample rate specified in RCO file”
6. Under Bandwidth and Timing change the Standard Sample Rate to 25kHz. (Ignore the Number of Time Slices).



Setup RP Device-A

RCO File

Standard Files: Select ...

File name: C:\TDT\SigGenRP\sgRCO\SwpPlay50Kx2

☐ Use sample rate specified in RCO file

Device Select

Type: RP2 Processor

Index: 1

Bandwidth and Timing

Standard Sample Rate: 25K

SR = 24414.0625

Number Time Slices: 10

The RPx device and index is the hardware device for playing out the signal. The RCO file is an RPvdsEx circuit that controls stimulus generation and presentation. The sample rate can be set to the standard rate by selecting “Use sample rate specifiecd…” Deselect “Use sample rate specified…” to choose a different sample period.

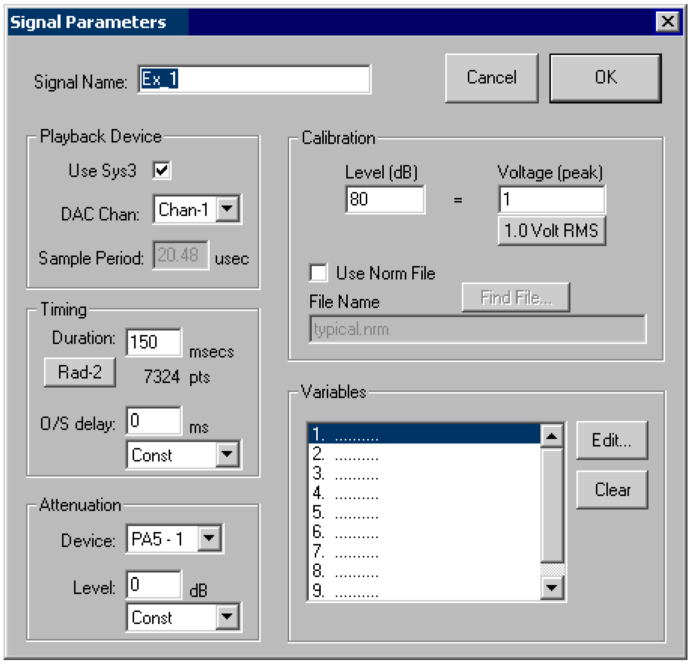
Define the Signal Parameters

In SigGen, a *signal* is defined as an overall waveform consisting of segments and components. These signal "building blocks" will be discussed later in this example. To create a signal in SigGen, you must first define the parameters for the overall waveform.

To define the signal parameters

1. Select Signal from the Modify menu of the main window, or click at the Signal icon on the toolbar.
2. Under Playback Device select “Use Sys3”.
3. Enter the following parameters in the Signal Parameters dialog box:

Name:	*Ex_1*
Duration:	150 msec
<u>Calibration</u>	
Level:	80 dB



Signal Name: Ex_1

Playback Device

Use Sys3 ☑

DAC Chan: Chan-1

Sample Period: 20.48 usec

Timing

Duration: 150 msecs

7324 pts

O/S delay: 0 ms

Const

Attenuation

Device: PA5 - 1

Level: 0 dB

Const

Calibration

Level (dB) 80 = Voltage (peak) 1

1.0 Volt RMS

☐ Use Norm File

File Name typical.nrm

Variables

1. ....
2. ....
3. ....
4. ....
5. ....
6. ....
7. ....
8. ....
9. ....

In this example, the signal duration is 150 msec and the sampling period is 20.48 µsec (the sample period text box is grayed out with System 3 hardware). The *sampling period* is the time interval between two digital signal points and is typically specified in microseconds (µsec). The *duration* parameter defines, in milliseconds (msec), how long the signal or segment will last. With a *sampling frequency* of 50 kHz (the inverse of the sampling period in seconds) and a signal duration of 150 msec, 7,324 points will be needed to generate the signal in this example.

In SigGen, the *calibration reference* defines the relationship between sound pressure level (in dB) and waveform amplitude (voltage level in Volts). The Level field in the Calibration group box is used to adjust the overall level of the digital segment generated by SigGen. All level measurements in SigGen are calculated in dB volts with a reference of 1 volt peak or 1 volt RMS (root mean squared). By setting the Level parameter to 80 dB in this example, the amplitude reference will be calibrated as 1 V peak = 80 dB. To express the calibration in actual decibel sound pressure level (dBSPL), measurement must be made for the system, and the dBSPL value corresponding to 1 volt peak signal must be entered in this field.

SigGen can produce signals in graphic and digital forms. In this example, the signal will be played in D/A channel 1. Since signal attenuation will not be needed for this signal, the Level parameter remains at zero.

To accept the parameters and return to the main menu

➢ Click the OK button.

Create the Segments and Components

A SigGen signal consists of one or more *segments*. These segments may be added or multiplied to produce the overall signal. Each SigGen segment consists of one to three *components*. The components are defined by either time or frequency domain parameters and are combined additively to produce a segment.

In this example, you will create one segment and one component for the single-tone blip signal.

To edit/create a segment

1. Select Segment from the Modify menu of the main window, or click at the Modify Segment icon on the toolbar.
2. Set the Duration parameter to 150 msec, or click the To Max button below the field to set the segment duration the same as the signal duration.

Leave the remaining parameters at their default settings. The parameters for the first segment will be:

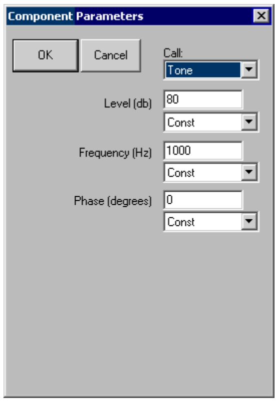
Select:	Seg[1]
Gate Type:	Cos^2
Gate Time:	5 msec
Level:	0.0 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	0.0 msec
Duration:	150 msec

To modify/create a component

1. Click one of the three buttons in the Components group box.
2. Enter the following (constant) parameters in the Component Parameters dialog box:

Call:	Tone
Level:	80 dB
Frequency:	1000 Hz

The Phase (cosine) parameter will remain at the default setting of 0.0 degrees.



Call: Tone

Level (db) 80 Const

Frequency (Hz) 1000 Const

Phase (degrees) 0 Const

The component parameters above define Seg[1] as a pure tone with an 80 dB amplitude, a 1000 Hz frequency, and a 0.0 degree starting phase. A *tone* is defined as a sinusoidal signal with only one frequency. In SigGen, there are three parameters to determine a tone: amplitude (or level), frequency, and phase. These parameters, as well as other related terms, will be introduced in later examples.

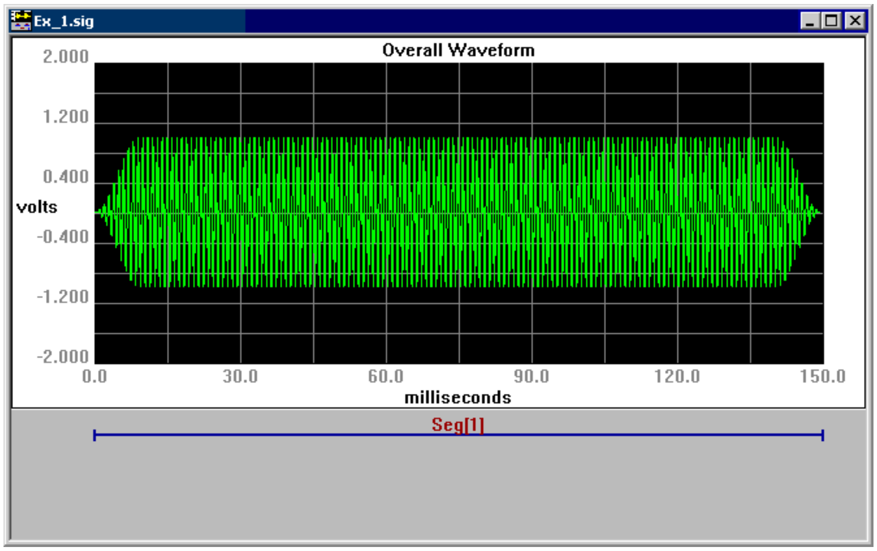
To accept the parameters and return to the Edit Signal Segments dialog box

- Click the OK button.

The dialog box above indicates that Seg[1] will be gated by a $\cos^2$ window with a rise and fall time of 5 msec. The only component of this segment will be generated in the time domain, starting at 0.0 msec and lasting for 150 msec, and will be added to the overall signal.

To accept the parameters and return to the main window

- Click the OK button.



Preview the Signal

To play the signal

➢ Click the Play button of the Signal Preview Area.

To stop signal play

➢ Click the Stop button.

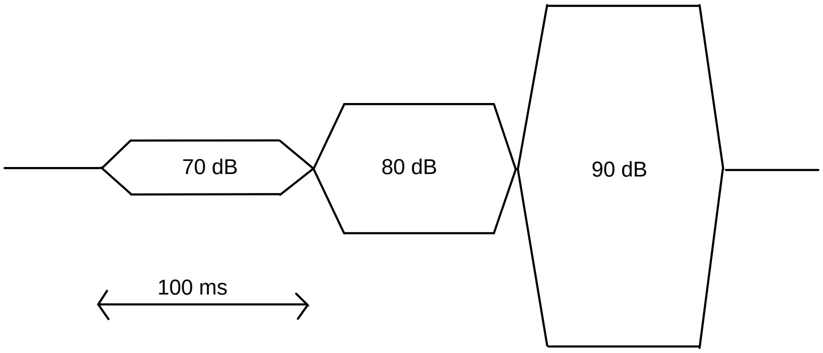
Save the Signal

SigGen allows you to save all of your signal parameters in a standard *SigGen file* format. The default extension for files saved in the SigGen format is *.sig*. You can also use any other extension.

To save the signal file

1. Select Save As... from the File menu of the main window.
2. Enter the desired file name, *Ex_1*.

Example 2: Generating a Three-Tone Blip Series



This example will show you how to generate a three-tone blip signal consisting of:

- A pure tone with a frequency of 500 Hz at a level of 70 dB, and gated by a cos^2 window;
- A pure tone with a frequency of 1500 Hz at a level of 80 dB, and gated by a cos^2 window; and
- A pure tone with its frequency changing on each trial. The frequency will step from 1500 Hz to 2500 Hz at a level of 90 dB.

The total duration of this signal will be 300 msec long, with each segment lasting 100 msec. The sampling frequency will be 100 kHz, which corresponds to a period of 10 µsec. The calibration reference will be 80 dB = 1.0 V peak voltage. Finally, this signal will be saved to a 16-bit binary file.

Throughout this example, several SigGen concepts will be discussed. These concepts are listed in the adjacent column, and will be marked by the SigGen icon when they are introduced.

Concepts introduced:

- ✓ *Variable*
- ✓ *SigGen Index*
- ✓ *SGI Modifiers*
- ✓ *Previewing Signal Intervals*
- ✓ *SigGen Wave File*
- ✓ *Gating*

To generate, preview, and save the three-tone blip signal, you will need to perform the following steps:

1. Run SigGen.
2. Define the signal parameters.
3. Create the segments and components.
4. Preview the signal.
5. Save the signal.

Note: Only those parameters to be entered or selected are given below. All other parameters will remain according to the default settings.

Run SigGen and Create a New Signal

To run SigGen

- Double-click the SigGen icon

  or

- Select SigGen from the Start\Program menu.

To create a new SigGen signal

- Select New from File menu, or click on the New Signal icon on the toolbar.

Configure the RPx device and RCO file

For System 3 equipment SigGenRP uses the RPx device and RCO files (See RPvdsEx help) to play out signals.

To configure the RPx device and select the RCO file

1. Select RP devices|Device A from the Modify menu or click on the "A" icon on the toolbar.
2. Select from the Standard file menu "Sweep Play, 2 Chan 50K (max 100K)".
3. Select RP2 Processor as device type
4. Select Index 1.
5. Deselect "Use sample rate specified in RCO file"
6. Under Bandwidth and Timing change the Standard Sample Rate to 25kHz. (Ignore the Number of Time Slices).

Define the Signal Parameters

To define the signal parameters

1. Select Signal from the Modify menu of the main window.
2. Enter the following parameters in the Signal Parameters dialog box:

Name:	*Ex_2*
Duration:	300 msec
Use Sys3:	Checked
<u>Calibration</u>	
Level:	80 dB

These parameters define a 300 msec long signal with a sampling period of 20.48 usec. By setting the Level parameter to 80 dB, the amplitude reference will be calibrated as 1 V peak = 80 dB.

Create a Variable

Since the third tone in this signal requires the frequency to step from 1500 to 2500 Hz, you will need to create a variable to generate such an effect.

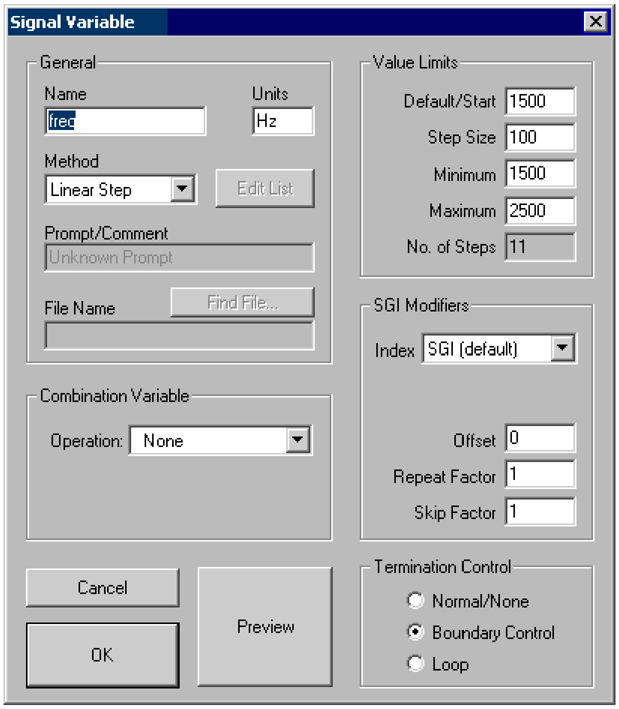
A SigGen *variable* allows you to change the value of signal and segment parameters. In SigGen, you can define the starting value of each variable, as well as establish signal boundaries by defining the maximum and minimum values. You can also specify the method in which these values will be varied.

This example illustrates just one way to control a variable (linear step). Later examples will describe other methods of controlling variables.

To create a variable

1. Click an empty variable field and the Edit button in the Variable box of the Signal Parameters dialog box.
2. Enter or select the following parameters in the Signal Variable dialog box:

Name:	*freq*
Method:	Linear Step
Value Limits	
Default/Start:	1500
Step:	100
Minimum:	1500
Maximum:	2500
Boundary Control:	☒
SGI Modifiers	
Offset:	0



Signal Variable

General

Name: freq

Units: Hz

Method: Linear Step

Prompt/Comment: Unknown Prompt

File Name:

Combination Variable

Operation: None

Value Limits

Default/Start: 1500

Step Size: 100

Minimum: 1500

Maximum: 2500

No. of Steps: 11

SGI Modifiers

Index: SGI (default)

Offset: 0

Repeat Factor: 1

Skip Factor: 1

Termination Control

○ Normal/None

● Boundary Control

○ Loop

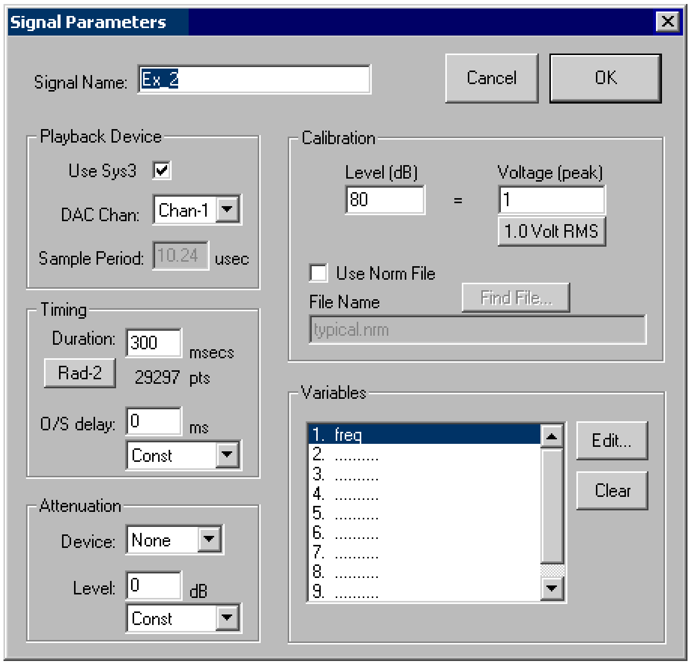
SigGen variables allow you to change the value of signal parameters as a function of the *SigGen Index (SGI)* value. The SGI indicates the current status of the signal. The linear step variable, *freq*, has a starting value of 1500 Hz and, for each SGI increment, will increase by a step of 100 Hz.

freq will be used to produce the increasing sweeping frequency for the third tone in your signal. Since *freq* is controlled by boundaries, the signal will automatically stop playing when it reaches the specified Maximum and/or Minimum value.

The parameters in the *SGI Modifiers* box allow you to define the starting point, the increment, and the number of intervals of SGI. With an Offset setting of 0, the initial value of *freq* will be 1500 Hz [1500 + 0(100)]. With a default setting of 1 for both the Repeat and Skip Factors, *freq* will increment each step by 100 Hz, starting at 1500 Hz.

To accept the parameters and return to the Signal Parameters dialog box

➢ Click the OK button in the Signal Variable dialog box.



Signal Parameters

Signal Name: Ex_2

Playback Device
Use Sys3 ☑
DAC Chan: Chan-1
Sample Period: 10.24 usec

Timing
Duration: 300 msecs
29297 pts
O/S delay: 0 ms
Const

Attenuation
Device: None
Level: 0 dB
Const

Calibration
Level (dB) 80 = Voltage (peak) 1
☐ Use Norm File
File Name typical.nrm

Variables
1. freq
2. ....
3. ....
4. ....
5. ....
6. ....
7. ....
8. ....
9. ....

To save the parameters and return to the main window

➢ Click the OK button in the Signal Parameters dialog box.

Create the Segments and Components

To create the first segment

1. Select Segment from the Modify menu of SigGen's main window.

2. Set the Duration parameter to 100 msec and leave the remaining parameters at their default settings.

The parameters for the first segment will be:

Select:	Seg[1]
Gate Type:	Cos^2
Gate Time:	5 msec
Level:	0.0 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	0.0 msec
Duration:	100 msec

To create a component for the first segment

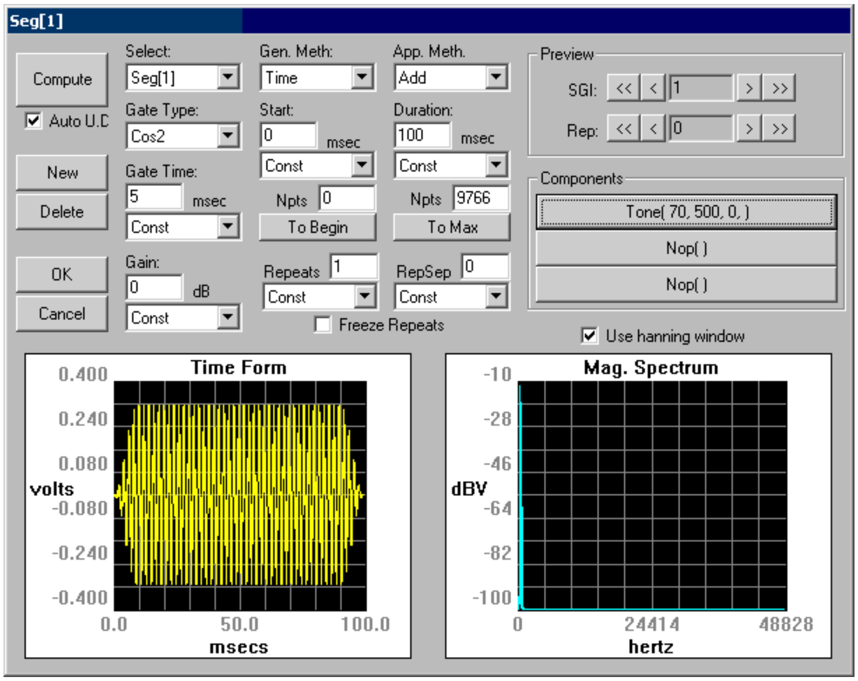
1. Click the first button in the Components group box.
2. Enter the following parameters in the Components Parameters dialog box:

Call:	Tone
Level:	70 dB
Frequency:	500 Hz

The Phase (cosine) parameter will remain at the default setting of 0.0 degrees.

To accept the parameters and return to the Edit Signal Segments dialog box

➢ Click the OK button.



Seg[1]

☑ Auto U.D

Select: Seg[1]

Gate Type: Cos2

Gate Time: 5 msec, Const

Gain: 0 dB, Const

Gen. Meth: Time

Start: 0 msec, Const

Npts 0

Repeats 1, Const

☐ Freeze Repeats

App. Meth: Add

Duration: 100 msec, Const

Npts 9766

RepSep 0, Const

Preview SGI: 1

Rep: 0

Components: Tone(70, 500, 0,) / Nop( ) / Nop( )

☑ Use hanning window

The parameters above indicate that Seg[1] and Seg[2] will be gated by a cos^2 window with a rise and fall time of 5 msec. In SigGen, a *gate* is defined as a window that modifies the rising and falling of a signal segment. SigGen offers four options: Cos^2, Hann, and Ramp gates, and None (no gate).

According to the parameters above, Seg[1] will have a tone component with a 70 dB amplitude, a 500 Hz frequency, and a 0.0 degree starting phase (default setting). It will be gated by a cos^2 window with a rise and fall time of 5 msec. The segment's only component will be generated in the time domain, starting at 0.0 msec and lasting for 100 msec, and will be added to the signal.

To create the second segment

1. Click the New button in the Edit Signal Segments dialog box.

2. Enter the following parameters for Seg[2]:

Start:	100 msec
Duration:	100 msec

The rest of the parameters for this segment are the default settings, and are the same as those for Seg[1].

To create a component for the second segment

1. Click the first button in the Components group box.

2. Enter the following parameters in the Components Parameters dialog box:

Call:	Tone
Level:	80 dB
Frequency:	1500 Hz

The Phase (cosine) parameter will remain at the default setting of 0.0 degrees.

To accept the parameters and return to the Edit Signal Segments dialog box

➢ Click the OK button.

To create the third segment

1. Click the New button in the Edit Signal Segments dialog box.

2. Enter the following parameters for Seg[3]:

Start:	200 msec
Duration:	100 msec

The rest of the parameters for this segment are the default settings, and are the same as those for Seg[1].

To create a component for the third segment

1. Click the first button in the Components group box.

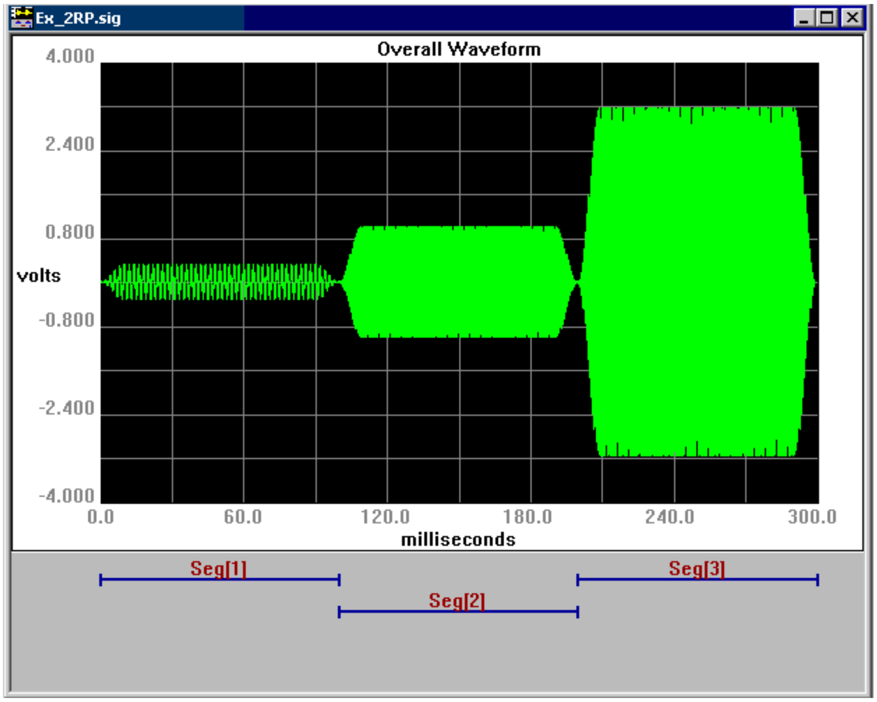
2. Enter the following parameters in the Components Parameters dialog box:

Call:	Tone
Level:	90 dB
Frequency:	*freq*

Then this third segment (tone) will have a frequency starting at 1500 Hz and incrementing by 100 Hz as SGI increments. The Phase (cosine) parameter for this segment will remain at the default setting of 0.0 degrees.

To accept the parameters and return to SigGen's main window

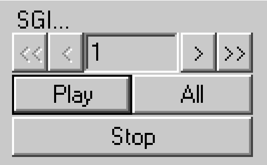
1. Click the OK button in the Component Parameters dialog box.
2. Click the OK button in the Edit Signal Segments dialog box.



According to the calibration, 80 dB = 1.0 V peak, the first segment is 70 dB, or 0.3162 V; the second segment is 80 dB, or 1.0 V; and the third segment is 90 dB, or 3.1623 V. Note that the signal shown above corresponds to these levels.

Preview the Signal

In many cases, one or more signal parameters will be varied from one interval to the next during an experiment. SigGen provides a means for previewing the signal through all of its intervals. You also may select a specific interval for preview.



To play all specified intervals of the display signal

- Click the All button.

To play a specific signal interval

1. Set the value of the SGI equal to the number you wish to preview.
2. Click the Play button of the Signal Preview Area.

To stop signal playing prior to the end of the signal

- Click the Stop button.

Save the Signal

SigGen allows you to save your signal in two different formats. The signal file saves all current SigGen settings. The *wave file* saves only the overall waveform in a standard 16-bit integer file for playing and processing. In this example, you will save the three-tone series signal in the wave file format.

To save the signal file

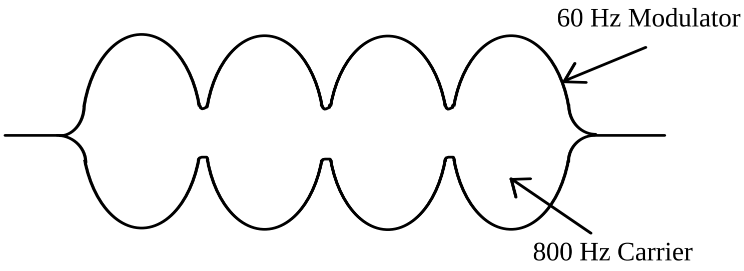
1. Select Save As... from the File menu of the main window.
2. Enter the desired file name, Ex_2.

To save the wave file

1. Select Save Wave from the File menu of the main window.
2. Enter the desired name for the 16-bit integer format file name, e.g., *Ex_2*.

SigGen will add the extension *.16* to the file name if you do not specify it. The *.16* file contains the generated waveform data for playing or further processing using SigGen or other software and hardware. Waveform files that are not generated by SigGen can be imported into SigGen by using a *.000* or a three-digit extension (refer to previous chapters). For usage of the 16-bit integer waveform file, please refer to TDT's software and hardware references.

Example 3: Generating an Amplitude-Modulated (AM) Signal



The waveform generated in this example will be an amplitude-modulated (AM) signal. The time method will be used to generate a waveform with a carrier frequency of 800 Hz and an amplitude modulated by 25% with a 60 Hz sinusoid. This is equivalent to a combination of three sinusoids with frequencies of 740 Hz, 800 Hz, and 860 Hz. The amplitude of the AM signal will be approximately 5.0 V, with a calibration of 80 dB = 1.0 V peak. The duration of this signal will be 100 msec.

Concepts introduced:

- ✓ *Application Methods*
- ✓ *AM Signal*

Depth of modulation is defined by subtracting the ratio of the valley and the peak amplitude of the AM signal from 100%. For example, the peak of the AM signal is 1Volt, and the valley is 0.25Volt, the ratio is 0.25, or 25%, and the depth of modulation is 75%.

Throughout this example, several SigGen concepts will be discussed. These concepts are listed in the adjacent column, and will be marked by the SigGen icon when they are introduced.

To generate the AM signal, you will need to perform the following steps:

1. Run SigGen.
2. Define the signal parameters.
3. Create the segments and components.

Note: Only those parameters to be entered or selected are given below. All other parameters will remain according to the default settings.

Run SigGen and Create a New Signal

To run SigGen

- ➢ Double-click the SigGen icon

  or

- ➢ Select SigGen from the Start\Program menu.

To create a new SigGen signal

- Select New from File menu, or click on the New Signal icon on the toolbar.

Configure the RPx device and RCO file

For System 3 equipment SigGenRP uses the RPx device and RCO files (see RPvdsEx help) to play out signals.

To configure the RPx device and select the RCO file

1. Select RP devices|Device A from the Modify menu or click on the "A" icon on the toolbar.
2. Select from the Standard file menu "Sweep Play, 2 Chan 50K (max 100K)".
3. Select RP2 Processor as device type
4. Select Index 1.
5. Deselect "Use sample rate specified in RCO file"
6. Under Bandwidth and Timing change the Standard Sample Rate to 25kHz. (Ignore the Number of Time Slices).

Define the Signal Parameters

To define the signal parameters

1. Select Signal from the Modify menu of the main window.
2. Enter the following parameters in the Signal Parameters dialog box:

Name:	*Ex_3*
Use Sys3:	checked
Duration:	100 msec
<u>Calibration</u>	
Level:	80 dB

In SigGen, there are two *Application Methods* for specifying how the segment(s) will construct the signal. The *Add* method adds the segment to the signal, whereas the *Multiply* method multiplies the segment to the signal.

In this example, you can generate the signal by using either of the two SigGen Application Methods. One way is to *multiply* the carrier by a modulation signal. The other is to *add* the carrier to lower and upper sideband signals. The amplitudes of the lower and upper sideband signals will be equivalent to one-fourth of the carrier amplitude. Their frequencies will differ from the carrier frequency by the modulation frequency. In this example, the Multiply method will be used to modulate the carrier signal.

Create the Segments and Components

When the amplitude of the AM signal is 5.0 V, and the depth of modulation is 25% (valley amplitude 3.75 V), the amplitude of the carrier must be 5.0 V, and the difference of the peak and valley of the modulator is 0.25 V and is elevated 0.75V (modulator amplitude is always 1.0 V). The 800 Hz carrier's level is 93.98 dB. The modulator will consists of three components: a tone and two DC Shifts. Since the modulator will be multiplied to the signal, component levels are in reference of 0 dB = 1.0 V. The level of the modulator tone is 0.125 V (half of 25% of 1.0V), that is 18 dB down from 1.0 V. Two DC Shifts will offsets the modulator tone by 0.125 V and 0.75 V separately, or -18 dB and -2.4988 dB.

To create the first segment

1. Select Segment from the Modify menu of the main window.
2. Set the Duration parameter to 100 msec and leave the remaining parameters at their default settings.

The parameters for the first segment will be:

Select:	Seg[1]
Gate Type:	None
Level:	0.0 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	0.0 msec
Duration:	100 msec

To create a component for the first segment

1. Click the first button in the Components group box.
2. Enter the following parameters in the Component Parameters dialog box:

Call:	Tone
Level:	93.98 dB (5.0 V)
Frequency:	800 Hz

The Phase (cosine) parameter will remain at the default setting of 0.0 degrees.

To create the second segment

1. Click the New button in the Edit Signal Segments dialog box.
2. Enter the following parameters for Seg[2]:

Gate Type:	None
Level:	0.0 dB
Gen. Meth.:	Time
App. Meth.:	Multiply
Start:	0.0 msec
Duration:	100 msec

The rest of the parameters for this segment are the same as those for Seg[1].

The Multiply Application Method chosen for the second segment indicates that the overall signal will be modulated by Seg[2]. The amplitude of the overall signal will change as the amplitude of the modulator changes — this modulation is called Amplitude Modulation (AM).

To create the first component for the second segment

1. Click the first button in the Components group box.
2. Enter the following parameters in the Component Parameters dialog box:

Call:	Tone
Level:	-18 dB
Frequency:	60 Hz

The Phase (cosine) parameter will remain at the default setting of 0.0 degrees.

To create the second component for the second segment

1. Click the second button in the Components group box.
2. Enter the following parameters in the Component Parameters dialog box:

Call:	DC Shift
Level:	-18 dB

Since the calibration reference for the components with Multiply application is based on 0 dB = 1 V, this level of DC Shift will elevate the first component by its level. Now the modulator varies between 0 and 0.25 V.

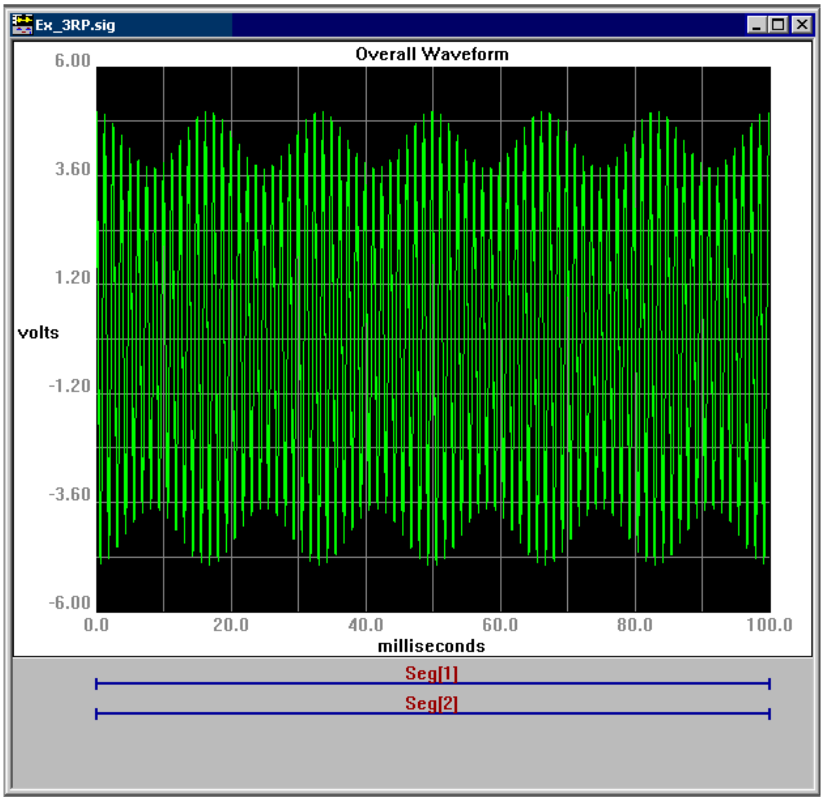
To create the third component for the second segment

1. Click the third button in the Components group box.
2. Enter the following parameters in the Component Parameters dialog box:

Call:	DC Shift
Level:	-2.4988 dB

As you can expect, -2.4988 dB will produce a DC shift of 0.75 V. Now as you can see in the Time Form plot, the modulator is completed.

The overall waveform, shown below, is composed of two multiplied segments. The effect of modulation can be seen clearly. Note that the signal is approximately 5.0 V, as designated at the beginning of this example.

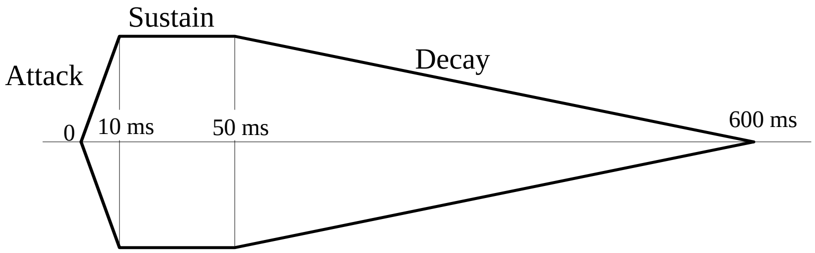


Save the Signal

To save the signal file

1. Select Save As... from the File menu of the main window.
2. Enter the desired file name, Ex_3.

Example 4: Generating a Multi-Component Signal



This example will show you how to generate 12 pseudo guitar notes across an octave, starting with the "A" note (110 Hz). Each note will include nine primary harmonics and three secondary harmonics. The full length of the signal will be 600 msec, with 10 msec of *attack*, 40 msec of *sustain*, and 550 msec of *decay*. The entire signal will be modulated by a frequency of 40 Hz and a depth of 10 percent. The output signal level will be approximately 5.0 V peak, with a calibration reference of 1.0 V = 80 dB.

This SigGen example is a complicated case of signal generation—multiple harmonics will be used to create the signal. In this example, pure tones will be used to represent the harmonics. The shaping envelope was generated using TDT's APPs software and is provided with your SigGen software.

Concepts introduced:

- ✓ *Variable file*
- ✓ *Using a Disk File (FileF)*

Throughout this example, several SigGen concepts will be discussed. These concepts are listed in the adjacent column, and will be marked by the SigGen icon when they are introduced.

This example is done in Time domain again. As one may expect, it can also be done in Frequency domain using SigGen Normalization file. The readers are encouraged to try the frequency method by themselves.

To generate this signal, you will need to perform the following steps:

1. Run SigGen.
2. Define the signal parameters.
3. Create a variable file.
4. Define the variables for each frequency.
5. Create the segments and components.
6. Create an envelope segment and apply it to the signal.

Note: Only those parameters to be entered or selected are given below. All other parameters will remain according to the default settings.

Run SigGen and Create a New Signal

To run SigGen

- Double-click the SigGen icon

  or

- Select SigGen from the Start\Program menu.

To create a new SigGen signal

- Select New from File menu, or click on the New Signal icon on the toolbar.

Configure the RPx device and RCO file

For System 3 equipment SigGenRP uses the RPx device and RCO files (see RPvdsEx help) to play out signals.

To configure the RPx device and select the RCO file

1. Select RP devices|Device A from the Modify menu or click on the "A" icon on the toolbar.
2. Select from the Standard file menu "Sweep Play, 2 Chan 50K (max 100K)".
3. Select RP2 Processor as device type
4. Select Index 1.
5. Deselect "Use sample rate specified in RCO file"
6. Under Bandwidth and Timing change the Standard Sample Rate to 25kHz. (Ignore the Number of Time Slices).

Define the Signal Parameters

With a calibration reference of 1.0 V = 80 dB and a 10% modulation, the 5.0 V peak output will correspond to a 93.98 dB base signal level.

To define the signal parameters

1. Select Signal from the Modify menu of the main window.
2. Enter the following parameters in the Signal Parameters dialog box:

Name:	*Ex_4*
Duration:	600 msec
<u>Calibration</u>	
Level:	80 dB

To accept the parameters and return to the main window

- Click the OK button.

Create the Variable File

The multi-component signal in this example will consist of multiple notes with varying frequencies. To define those frequencies, you will need to use a variable file. In SigGen, a *variable file* is an ASCII text file that contains the names and values for user-customized variables. Variable files may be created with any text editor, such as Windows Notepad. For more information about the format of SigGen's variable files, please refer to "Preparing to Build a Signal - The First Step" of this guide.

The variable file, *ex_4v.sch*, is provided with your SigGen software, and may be viewed in Notepad. It contains the harmonic frequencies for each of the 12 guitar notes. The 10 frequencies of each note are named *f1*, *f2*, *f3*, ..., *f9*, *s1*. The first frequency, *f1*, is defined as follows:

```
The first frequency of the primary
harmonics
{
f1
1
110.0
116.541
123.471
130.813
138.591
146.832
155.563
164.814
174.614
184.997
195.998
207.652
}
```

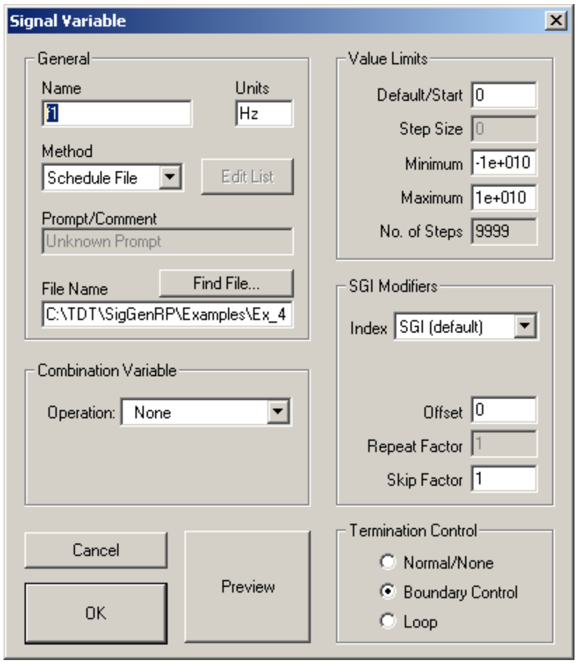
Define the Variables for Each Frequency

Although the variables *f1, f2, ..., f9, s1* are already defined in the *ex_4v.sch* file, you will still have to define each of them in SigGen.

To define the variables for f1

1. Select Signal from the Modify menu of the main window.
2. Click the Edit button in the Variables group box of the Signal Parameters dialog box.
3. Enter or select the following parameters in the Signal Variable dialog box:

Name:	*f1*
Method:	File
File Name:	*C:\TDT\SigGenRP\Examples\EX_4V.SCH*
Termination Control:	Boundary Control


Signal Variable

General
Name: f1
Units: Hz
Method: Schedule File
Prompt/Comment: Unknown Prompt
File Name: C:\TDT\SigGenRP\Examples\Ex_4

Combination Variable
Operation: None

Value Limits
Default/Start: 0
Step Size: 0
Minimum: -1e+010
Maximum: 1e+010
No. of Steps: 9999

SGI Modifiers
Index: SGI (default)
Offset: 0
Repeat Factor: 1
Skip Factor: 1

Termination Control
○ Normal/None
● Boundary Control
○ Loop

When entering the File Name, you must also provide the full path of the file name, or use the Find File ... button to locate the file. The variable name must match exactly the variable name defined in the variable file.

Note that variable *f1* varies according to the values in the *ex_4v.sch file* and the SGI value.

To accept the parameters and return to the Signal Parameters dialog box

- Click the OK button in the Signal Variable dialog box.

To define the variables for f2, f3, ..., f9, s1

1. For each frequency, select the next available variable in the Variables group box of the Signal Parameters dialog box.
2. Click the Edit button.
3. For each frequency, enter the following parameters in the Signal Variable dialog box.

Name:	*f2*, *f3*, *f4*, *f5*, *f6*, *f7*, *f8*, *f9*, or *s1*
Method:	File
File Name:	*C:\TDT\SigGenRP\Examples\EX_4V.SCH*

Create the Segments and Components

This signal will consist of six segments, all having the same duration. The first three segments will be the primary harmonics, the fourth will be the secondary harmonics, the fifth will be the modulation tone, and the sixth will be the envelope. The normalized levels of each component are listed below.

Name	**dB Level**	**Voltage**
f1	0	1.0
f2	-25	0.0562
f3	-15	0.2512
f4	-35	0.0178
f5	-45	0.0056
f6	-40	0.01
f7	-47	0.0045
f8	-45	0.0056
f9	-47	0.0045
s1	-20	0.1

The total voltage level will be 1.4554 V peak (3.2596 dB) for these normalized components. In order to get a 5.0 V peak signal, the actual total voltage level will need to be 5.0 V; that is, a 90.7198 dB gain will be needed.

To create the first segment

1. Select Segment from the Modify menu of the main window.
2. Enter or select the following parameters in the Edit Signal Segments dialog box:

Gate Type:	None
Gate Time:	5 msec
Level:	90.7198 dB
Duration:	600 msec

The parameters for the first segment will be:

Select:	Seg[1]
Gate Type:	None
Gate Time:	5 msec (dimmed)
Level:	90.7198 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	0.0 msec
Duration:	600 msec

To create three components for the first segment

1. Click the first button in the Components group box.
2. For each of the three components, enter or select the following parameters in the Component Parameters dialog box:

Component	**Call**	**Level (dB)**	**Frequency (Hz)**	**Phase (Degree)**
First	Tone	0.0	*f1*	0.0
Second	Tone	-25.0	*f2*	0.0
Third	Tone	-15.0	*f3*	0.0

To accept the parameters and return to the Edit Signal Segments dialog box

➢ Click the OK button.

To create the second through fifth segments and their corresponding components

1. Click the New button in the Edit Signal Segments dialog box.
2. Create the segments and components according to the following parameters.

<u>Second segment parameters:</u>

Select:	Seg[2]
Gate Type:	None
Gate Time:	5 msec
Level:	90.7198 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	0.0 msec
Duration:	600 msec

<u>Second segment components:</u>

Tone(-35.0, *f4*, 0)

Tone(-45.0, *f5*, 0)

Tone(-40.0, *f6*, 0)

These three tones will be for the fourth, fifth, and sixth primary harmonics.

<u>Third segment parameters:</u>

Select:	Seg[3]
Gate Type:	None
Gate Time:	5 msec
Level:	90.7198 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	0.0 msec
Duration:	600 msec

Third segment components:

Tone(-47.0, *f7*, 0)

Tone(-45.0, *f8*, 0)

Tone(-47.0, *f9*, 0)

These three tones will be for the seventh, eighth , and ninth primary harmonics.

Fourth segment parameters:

Select:	Seg[4]
Gate Type:	None
Gate Time:	5 msec
Level:	90.7198 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	0.0 msec
Duration:	600 msec

This segment will be the secondary harmonics.

Fourth segment component:

Tone(-20.0, *s1*, 0)

Fifth segment parameters:

Select:	Seg[5]
Gate Type:	None
Gate Time:	5 msec
Level:	0.0 dB
Gen. Meth.:	Time
App. Meth.:	Multiply
Start:	0.0 msec
Duration:	600 msec

This segment will be the modulator. The level of this modulator will be 0.5 V peak (-6.0205 dB), with a DC shift of 0.5 V to maintain the signal level at a depth of 10% (-0.91515 dB).

Fifth segment component:

Tone(-6.0205, 40, 0)
DC Shift(-6.0205)

Create an Envelope

Since the overall waveform in this example has a signal duration of 600 msec and a sampling period of 40 µsec, the shaping envelope will have to consist of 15,000 points. The envelope will increase linearly from 0 to 1.0 in the first 250 points, remain at 1.0 for 1000 points, and then decrease from 1.0 to 0 for the remaining 13,750 points.

In SigGen, complicated signals can be created by reading *disk files*. SigGen has two types of file formats, a 16-bit integer file (*File16*) and a floating-point binary file (*FileF*). SigGen signal files should have a 3-digit extension. This extension can be used with variables to select files.

The disk file, *env.000*, will be used in this example and is provided with your SigGen software. It is in the standard floating-point binary format and was generated using TDT's APPs software. Other methods also can be used to generate this file.

As a recommended exercise, the readers can generate an envelope file using 16-bit integer binary format (File16) instead of FileF for the following segment.

To create an envelope segment

1. Select Segment from the Modify menu of the main window.
2. Click the New button in the Edit Signal Segments dialog box.
3. Create the sixth segment according to the following parameters:

Select:	Seg[6]
Gate Type:	None
Gate Time:	5 msec
Level:	0.0 dB
Gen. Meth.:	Time
App. Meth.:	Multiply
Start:	0.0 msec
Duration:	600 msec

To create a component for the sixth segment

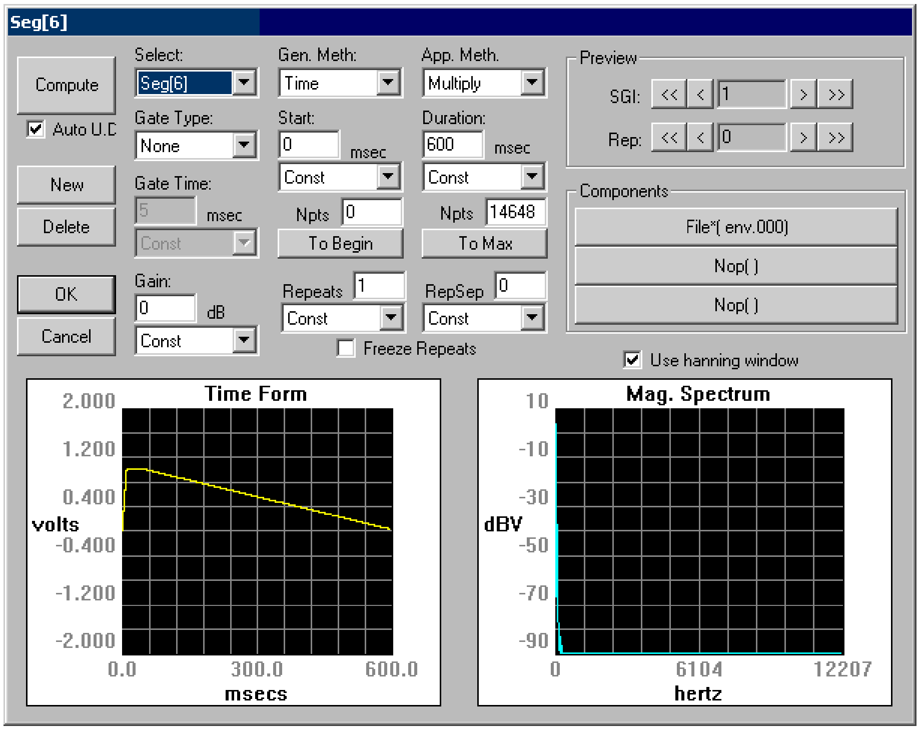
1. Click the first button in the Components group box.
2. Enter or select the following parameters in the Component Parameters dialog box:

Call:	FileF
File Name:	*env.000*

This binary data file must be in the current directory where SigGen is working.

To accept the component parameters and return to the Edit Signal Segments dialog box

➢ Click OK in the Component Parameters dialog box.



Seg[6]

Compute

☑ Auto U.D

New

Delete

OK

Cancel

Select: Seg[6]

Gate Type: None

Gate Time: 5 msec, Const

Gain: 0 dB, Const

Gen. Meth: Time

Start: 0 msec, Const

Npts 0

To Begin

Repeats 1, Const

App. Meth: Multiply

Duration: 600 msec, Const

Npts 14648

To Max

RepSep 0, Const

□ Freeze Repeats

Preview

SGI: << < 1 > >>

Rep: << < 0 > >>

Components

File*(env.000)

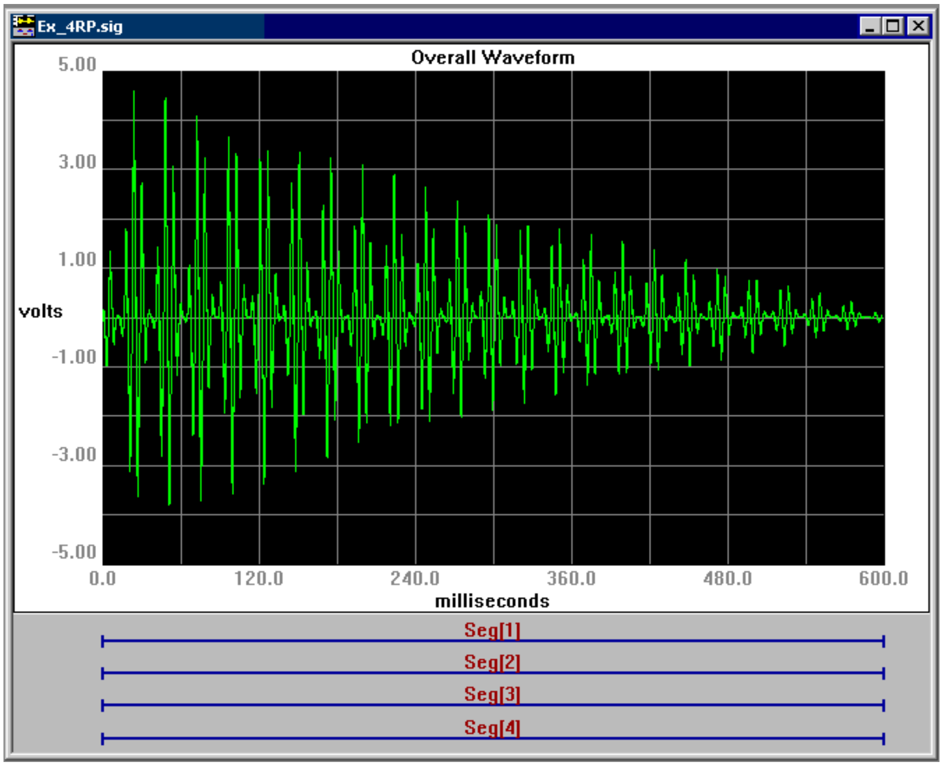
Nop( )

Nop( )

☑ Use hanning window

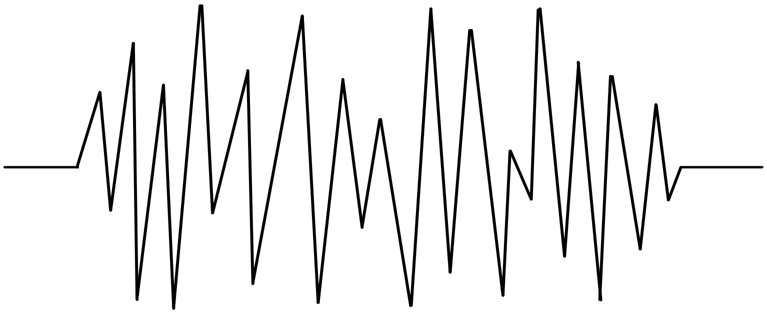
To accept the segment parameters and return to the main window

➢ Click the OK button.



As one may notice that, due to the cancellation of individual components, the final signal amplitude is not exactly 5 V. To make it exactly 5 V, you can modify the envelope file by changing 1.0 to 1.25.

Example 5: Generating a Band-Limited Gaussian Noise



Limited-Band Gaussian Noise, 500 msec Duration

Concepts introduced:

- ✓ *Frequency Method*
- ✓ *Segment Level*
- ✓ *Variable File*
- ✓ *Normalization File*

In this example, you will create a series of 500-msec, band-limited noise bursts with a 2-octave band starting with a center frequency of 600 Hz and stepping to 2000 Hz at a step size of 100 Hz. A 20-msec cos^2 gate will be applied to the signal. The output signal level is to be held at 2.0Vrms. A normalization file will be used to modify the magnitude and phase of the components in different frequency ranges.

Throughout this example, several SigGen concepts will be discussed. These concepts are listed in the adjacent column, and will be marked by the SigGen icon when they are introduced.

To generate the band-limited Gaussian noise, you will need to perform the following steps:

1. Create a normalization file.
2. Run SigGen.
3. Define the signal parameters.
4. Create the segments and components.
5. Add Norm file to the signal.

Note: Only those parameters to be entered or selected are given below. All other parameters will remain according to the default settings.

Create a Normalization File

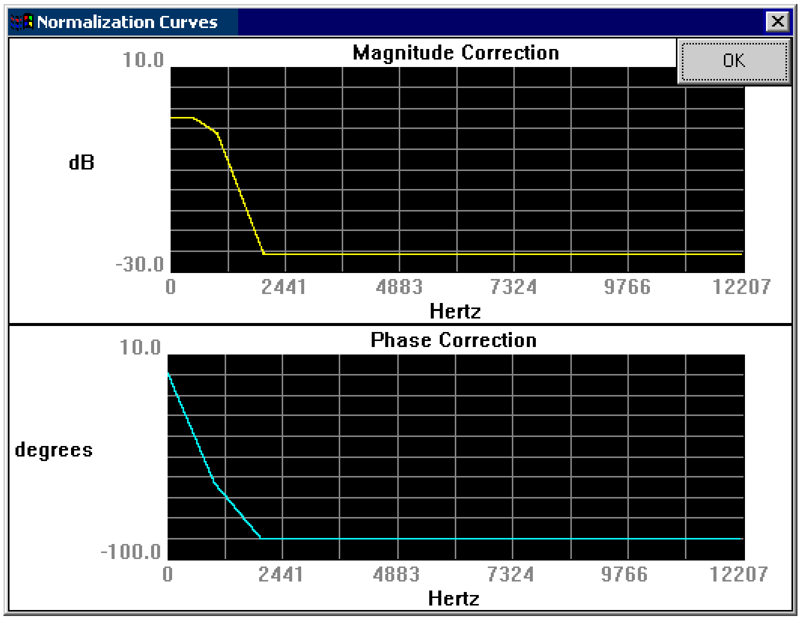
In SigGen, a *normalization file* (*.nrm* file) is a text file that states the corrections of amplitude (in dB) and starting phase (in degrees) of specific frequency values. SigGen will use the data provided in this file to normalize the output signal. This frequency normalization will provide a way to calibrate signals consisting of different frequency components. For more information about the format of SigGen's normalization files, please refer to "Preparing to Build a Signal - The First Step" of this guide.

When the normalization file is defined, the normalization curve can be displayed by clicking the H_f button in the SigGen main menu toolbar.

The normalization file, *ex_5.nrm*, can be created using a text editor. The file is defined as follows:

```
0.0, 0.0, 0.0
500.0, 0.0, -30.0
1000.0, -3.0, -60.0
2000.0, -27.0, -90.0
3000.0, -51.0, -120.0
```

This file defines a frequency-wise normalization, as illustrated below:



This normalization corrects each of the signal components by the normalization magnitude and phase. For instance, the signal component at 2000 Hz will be delayed in phase by 90 degrees, and reduced in magnitude by 27 dB.

Run SigGen and Create a New Signal

To run SigGen

- Double-click the SigGen icon

  or

- Select SigGen from the Start\Program menu.

To create a new SigGen signal

- Select New from File menu, or click on the New Signal icon on the toolbar.

Configure the RPx device and RCO file

For System 3 equipment SigGenRP uses the RPx device and RCO files (see RPvdsEx help) to play out signals.

To configure the RPx device and select the RCO file

1. Select RP devices|Device A from the Modify menu or click on the “A” icon on the toolbar.
2. Select from the Standard file menu “Sweep Play, 2 Chan 50K (max 100K)”.
3. Select RP2 Processor as device type
4. Select Index 1.
5. Deselect “Use sample rate specified in RCO file”
6. Under Bandwidth and Timing change the Standard Sample Rate to 25kHz. (Ignore the Number of Time slices).

Define the Signal Parameters

To define the signal parameters

1. Select Signal from the Modify menu of the main window.
2. Enter the following parameters in the Signal Parameters dialog box:

Name:	*Ex_5*
Duration:	500 msec
<u>Calibration</u>	
Level:	80 dB
Voltage:	1.41421 Volt-peak (1.0 Volt RMS)

The following may be applied later after viewing the signal without the effects of the normalization file:

Use Norm File	☒
File Name:	*C:\TDT\SigGenRP\Examples\EX_5.NRM*

Two SigGen variables, *freq* and *level* will be created so that we can step through increasing values of center frequency.

To create the variable freq

1. Click the first position and then the Edit button in the Variable group box of the Signal Parameters dialog box.

2. Enter or select the following parameters in the Signal Variable dialog box:

Name:	*freq*
Method:	Linear Step
<u>Value Limits</u>	
Default/Start:	600
Step:	100
Minimum:	500
Maximum:	2000
Boundary Control:	☒

To create the variable level

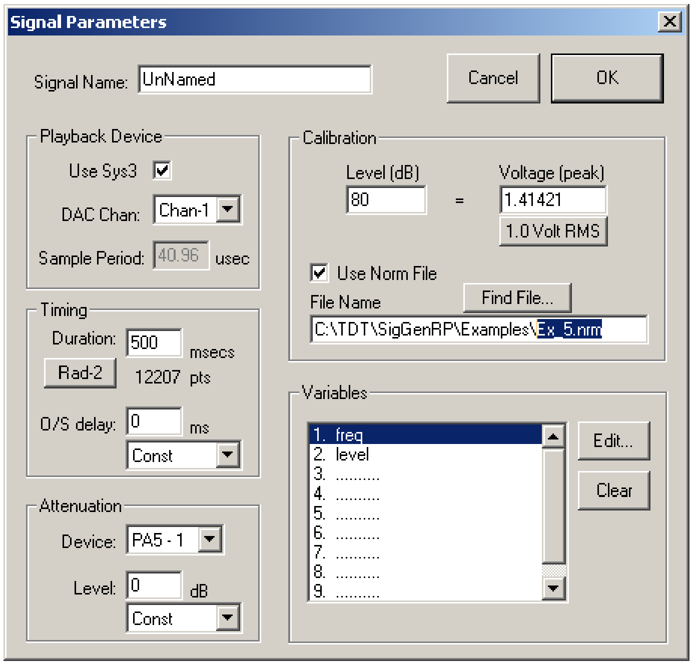
1. Click the second position and then the Edit button in the Variable group box of the Signal Parameters dialog box.

2. Enter or select the following parameters in the Signal Variable dialog box:

Name:	*level*
Method:	File
File Name:	C:\TDT\SigGenRP\Examples\EX_5V.SCH

To accept the parameters and return to the Signal Parameters dialog box

➢ Click the OK button in the Signal Variable dialog box.



Signal Name: UnNamed

Use Sys3 ☑

DAC Chan: Chan-1

Sample Period: 40.96 usec

Duration: 500 msecs

Rad-2 12207 pts

O/S delay: 0 ms

Const

Device: PA5 - 1

Level: 0 dB

Const

Level (dB) 80 = Voltage (peak) 1.41421

1.0 Volt RMS

☑ Use Norm File

File Name C:\TDT\SigGenRP\Examples\Ex_5.nrm

1. freq
2. level
3. ....
4. ....
5. ....
6. ....
7. ....
8. ....
9. ....

Create the Segment and Component

There will be only one segment for this signal. The frequency method will be used to generate the band-limited noise.

To create the segment

1. Select Segment from the Modify menu of the main window.
2. Enter or select the following parameters:

Select:	Seg[1]
Gate Type:	Cos^2
Gate Time:	20 msec
Level:	80 dB
Gen. Meth.:	Freq.(8192)
App. Meth.:	Add
Start:	0.0 msec (Begin)
Duration:	500 msec (Max.)

The *frequency method* in SigGen specifies a spectrum first, and then generates the signal component using an inverse FFT. The number of magnitude points determines the frequency resolution. In this example, the frequency value is 8192, which corresponds to 4096 magnitude points and 4096 phase points.

The Level parameter in the Edit Signal Segments dialog box specifies the attenuation or amplification applied to the components of the segment. This is just one of the many ways to regulate signal components in SigGen.

To create a component

1. Click the first button in the Components group box.
2. Enter the following parameters in the Component Parameters dialog box:

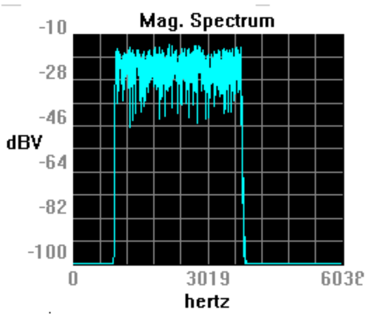
Call:	Band
Low Freq. Level:	*level*
High Freq. Level:	*level*
Center Freq.:	*freq*
Band Width (Oct.):	2
Roll-off (dB/oct.):	0

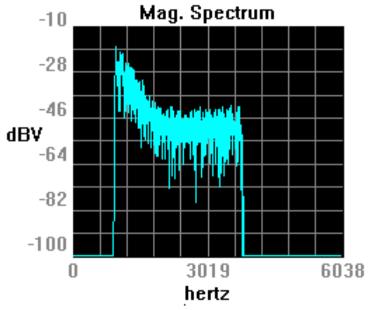
As you may remember, a value 0 in the Roll-off field will result in the maximum roll-off which SigGen can generate.

To accept the parameters and return to the Edit Signal Segments dialog box

➢ Click OK in the Components Parameters box.

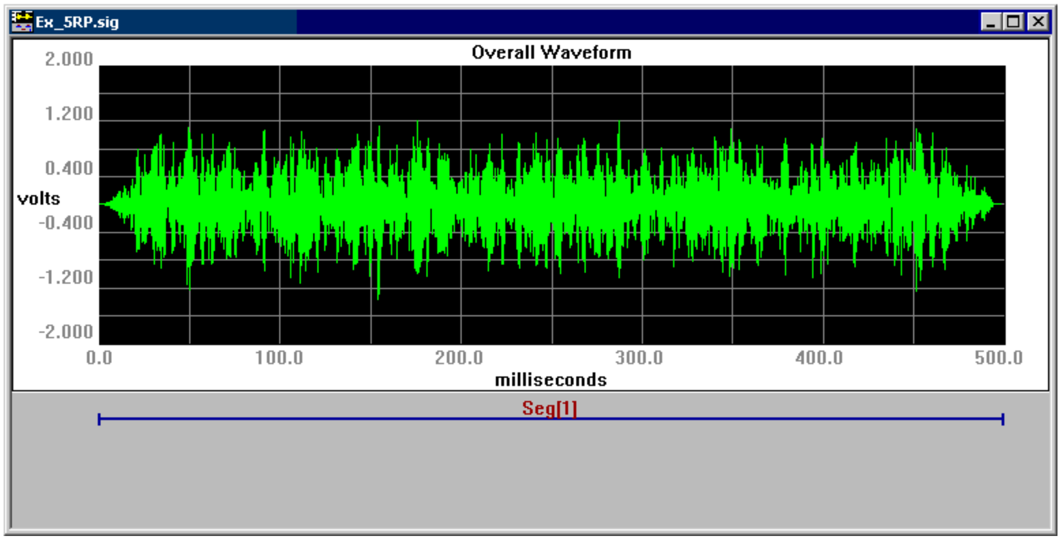
At this stage, after you view the Magnitude Spectrum in the Edit Segment Parameters window, you can return to Modify Signal Parameters window to enter the normalization file to the signal. The effect of normalization (frequency-domain correction) is illustrated by the two magnitude spectrum plots below.





The first spectrum shows the signal without normalization. Its level at 2000 Hz is about -10.0 dB. The second plot is the spectrum modified by the normalization file. Note that the signal level across the noise band is gradually reduced by 20 dB.

The band-limited noise is shown below:



Chapter 9 Examples: Speech Sciences

In this chapter, examples of speech signal processing are presented. Each example introduces new SigGen concepts. The examples found in this chapter are listed below. Also listed are the concepts introduced in each example.

Example 6: Creating Speech in Noise

- Using a Disk File (File16)
- Printing the Signal

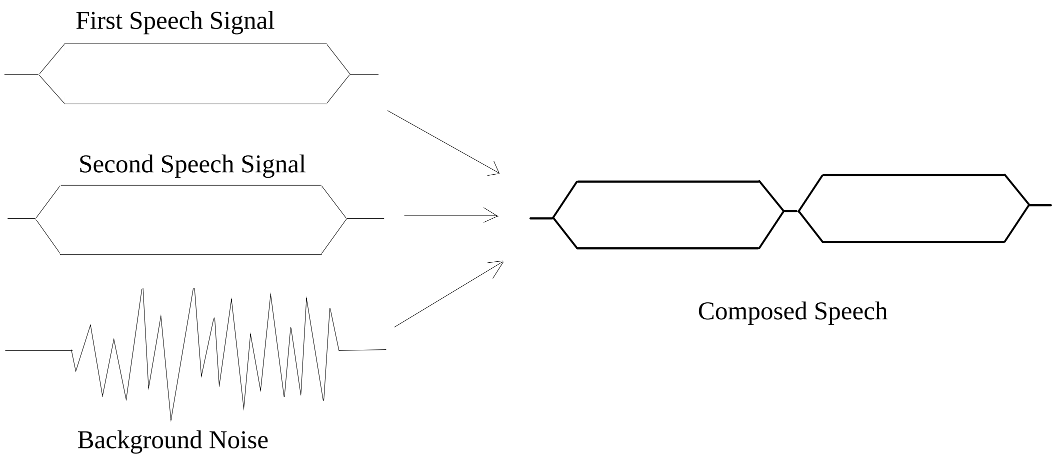
Example 7: Approximating A Glottal Waveform

- Applying a Spectral Envelope
- Using Normalization File
- Zoom-in
- Using Harm

Example 8: Producing a Steady State Vowel

- Using a Hanning window
- Saving a signal for later use as an imported component

Example 6: Creating Speech in Noise



Concepts introduced:

- ✓ *Using a Disk File (File16)*
- ✓ *Printing the Signal*

In this example, you will read from disk and concatenate two long speech files. These signals are stored in 16-bit integer binary format. You will then add low-level Gaussian noise. The level of the noise will be varying to show the effect of background noise.

Throughout this example, several SigGen concepts will be discussed. These concepts are listed in the adjacent column, and will be marked by the SigGen icon when they are introduced.

To generate the speech in noise, you will need to perform the following steps:

1. Create the speech files.
2. Run SigGen.
3. Define the signal parameters.
4. Create the segments and components.
5. Print the signal.

Note: Only those parameters to be entered or selected are given below. All other parameters will remain according to the default settings.

Create the Speech Files

In SigGen, there are two types of files that can be used for input as a signal component: FileF and File16. FileF is a floating-point binary file (see "Example 4: Generating a Multi-Component Signal"). *File16* is a 16-bit integer binary file with no header. When a File16 file is read in as a time signal, SigGen will convert it to a ±10 V waveform (corresponding to +32767 and -32767).

A File16 speech file can be created by recording a speech signal directly to a file using many commercial sound recording systems, or TDT's systems. In this example, the two files are two simple speech files. The first speech is 1.8 seconds long, and the second 3.0 seconds. The speech is recorded at a sampling rate of 25 kHz.

Run SigGen and Create a New Signal

To run SigGen

- ➢ Double-click the SigGen icon

  or

- ➢ Select SigGen from the Start\Program menu.

To create a new SigGen signal

- ➢ Select New from File menu, or click on the New Signal icon on the toolbar.

Configure the RPx device and RCO file

For System 3 equipment SigGenRP uses the RPx device and RCO files (see RPvdsEx help) to play out signals.

To configure the RPx device and select the RCO file

1. Select RP devices|Device A from the Modify menu or click on the "A" icon on the toolbar.
2. Select from the Standard file menu "Sweep Play, 2 Chan 50K (max 100K)".
3. Select RP2 Processor as device type
4. Select Index 1.
5. Deselect "Use sample rate specified in RCO file"
6. Under Bandwidth and Timing change the Standard Sample Rate to 25kHz. (Ignore the Number of Time slices).

Define the Signal Parameters

To define the signal parameters

1. Select Signal from the Modify menu of the main window.
2. Enter or select the following parameters in the Signal Parameters dialog box:

Name:	*Ex_6*
Duration:	4800 msec
<u>Calibration</u>	

Level:	80 dB

Note: Due to the length of the speech signal, you must have 8 Meg AP2 memory to exercise this example.

A variable, *noise_level*, will be created and used to control the band-limited Gaussian noise level. There are only two values for this noise: 0dB and 70dB (with and without noise).

To create the variable noise_level

1. Click the first position and then the Edit button in the Variable group box of the Signal Parameters dialog box.
2. Enter or select the following parameters in the Signal Variable dialog box:

Name:	*noise_level*
Method:	Linear Step
Value Limits	
Default/Start:	0
Step:	70
Minimum:	0
Maximum:	70
Boundary Control:	☒

Create the Segments and Components

The overall waveform in this example will consist of three segments. The time method will be used for all of these segments.

To create the first segment

1. Select Segment from the Modify menu of the main window.
2. Set the Duration parameter to 1800 msec and leave the remaining parameters at their default settings.

The parameters for the first segment will be:

Select:	Seg[1]
Gate Type:	Cos^2
Gate Time:	20 msec
Level:	0.0 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	0.0 msec
Duration:	1800 msec

To create a component for the first segment

1. Click the first button in the Components group box.
2. Enter or select the following parameters in the Component Parameters dialog box:

Call:	File16
File Name:	*EX_6S1.000*

To create the second and third segments and their components

1. Select Segment from the Modify menu of the main window.
2. Set the following parameters for each segment:

Second segment parameters:

Select:	Seg[2]
Gate Type:	Cos^2
Gate Time:	20 msec
Level:	0.0 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	1800 msec
Duration:	3000 msec

Second segment component:

Call:	File16
File Name:	*EX_6S2.000*

Third segment parameters:

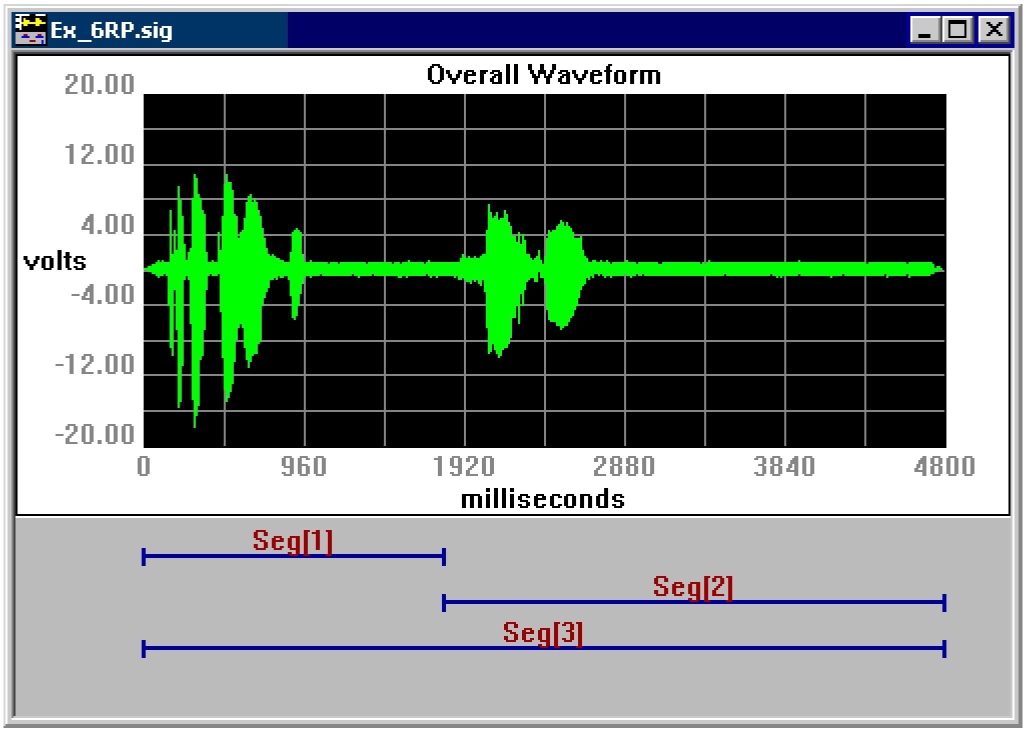
Select:	Seg[3]
Gate Type:	Cos^2
Gate Time:	20 msec
Level:	0.0 dB
Gen. Meth.:	Time
App. Meth.:	Add
Start:	0.0 msec
Duration:	4800 msec

Third segment component:

Call:	Gauss
Level:	*noise_level*

To return to SigGen's main window

➢ Click OK in the Edit Signal Segments dialog box.



Print the Signal

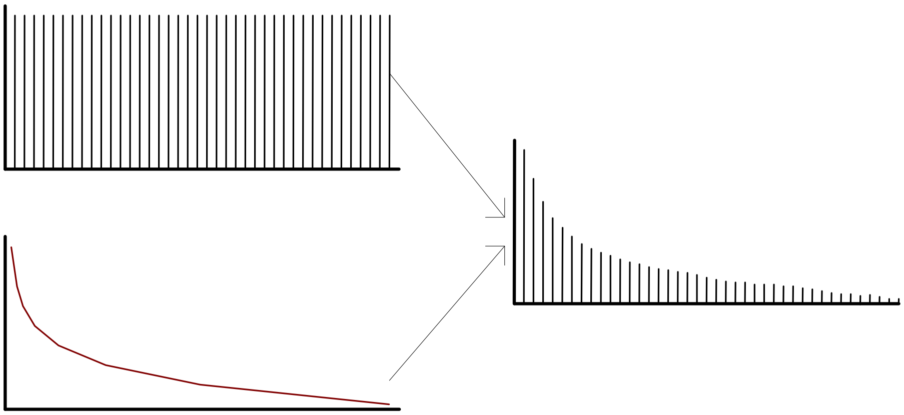
In SigGen, a signal waveform can be printed directly through a Windows printer. SigGen has three printing options: Print, Print Preview, and Print Setup.

To print the signal

➢ Select Print under the File menu of the main window.

The current signal setup will be printed with the waveform.

Example 7: Approximating A Glottal Waveform



Concepts introduced:

- ✓ *Applying a Spectral Envelope*
- ✓ *Using Harm*
- ✓ *Using Normalization File*

In this example, you will create an approximation of a glottal waveform. The *Harm* option will be used to generate a signal consisting of energy at the fundamental frequency and its integer multiples. Level will remain constant across harmonics. A normalization file that reduces level at a rate of 12 dB per octave will then be applied to the original signal, resulting in a signal consisting of energy at the fundamental frequency and its integer multiples, the level of which falls off at a rate of 12 dB per octave.

Throughout this example, several SigGen concepts will be discussed. These concepts are listed in the adjacent column, and will be marked by the SigGen icon when they are introduced.

To generate the glottal waveform, you will need to perform the following steps:

1. Create a normalization file.
2. Run SigGen.
3. Define the signal parameters.
4. Create the segment and component.
5. Zoom-in.
6. Save the signal as a SigGen signal and as a binary waveform.

Note: Only those parameters to be entered or selected are given below. All other parameters will remain according to the default settings.

Create a Normalization File

A normalization file (*.nrm* files) is an ASCII text file that specifies correction factors for amplitude (in dB) and phase (in degrees) at specific frequencies. The normalization file created for this example will be used to *apply a spectral envelope* that will reduce amplitude at a rate of approximately 12 dB per octave. No corrections will be specified for phase. For more information about the format of SigGen's normalization files, please refer to "Preparing to Build a Signal - The First Step" of this guide.

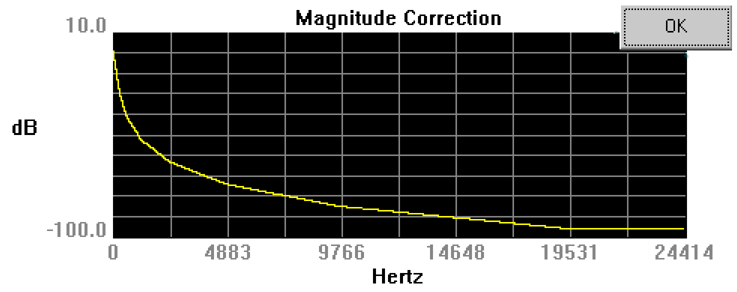
The variable file, *ex_7.nrm*, is provided with your SigGen software, and may be viewed in Notepad. You can create this file using a text editor. The file is defined as follows:

```
0.0, 0.0, 0.0
150.0, -12.0, 0.0
300.0, -24.0, 0.0
600.0, -36.0, 0.0
1200.0, -48.0, 0.0
2400.0, -60.0, 0.0
4800.0, -72.0, 0.0
9600.0, -84.0, 0.0
19200.0, -96.0, 0.0
```

The above normalization file chooses an arbitrary starting frequency of 150 Hz from which to build a spectral envelope that approximates a fall off of 12 dB per octave. The first parameter specifies frequency. The second parameter specifies the amount of amplitude reduction (in dB) to be applied at the specified frequency. The third parameter specifies phase correction. Notice that as frequency is doubled, amplitude is reduced by an additional 12 dB.

SigGen reads the information stored in the normalization file and uses it to correct amplitude. The correction factors specified for each discrete frequency are used to generate a continuous amplitude correction curve. This is accomplished by applying a linear function to each pair of successive correction factor values. Thus, the correction curve generated by the above file is not truly curvilinear, but consists of 8 linear segments that approximate a fall off of 12 dB per octave.

The resulting normalization curve is seen below:



Run SigGen and Create a New Signal

To run SigGen

- ➢ Double-click the SigGen icon

or

- ➢ Select SigGen from the Start\Program menu.

To create a new SigGen signal

- ➢ Select New from File menu, or click on the New Signal icon on the toolbar.

Configure the RPx device and RCO file

For System 3 equipment SigGenRP uses the RPx device and RCO files (see RPvdsEx help) to play out signals.

To configure the RPx device and select the RCO file

1. Select RP devices|Device A from the Modify menu or click on the "A" icon on the toolbar.
2. Select from the Standard file menu "Sweep Play, 2 Chan 50K (max 100K)".
3. Select RP2 Processor as device type
4. Select Index 1.
5. Deselect "Use sample rate specified in RCO file"
6. Under Bandwidth and Timing change the Standard Sample Rate to 25kHz. (Ignore the Number of Time slices).

Define the Signal Parameters

To define the signal parameters

1. Select Signal from the Modify menu of the main window.
2. Enter or select the following parameters in the Signal Parameters dialog box:

Name:	*Ex_7*
Duration:	200 milliseconds
Sample Period:	25 µseconds
<u>Calibration</u>	
Level:	80 dB
Use Norm File	☒
File Name	*C:\TDT\SigGenRP\Examples\EX_7.NRM*

In this example, the signal will be low-pass filtered such that the cut-off frequency is at or below 10,000 Hz. Therefore, we have increased the sample period from the default of 20 µ seconds to 50 µseconds.

The normalization file, *ex_7.nrm*, is defined as:

```
0.0, 0.0, 0.0
150.0, -12.0, 0.0
300.0, -24.0, 0.0
600.0, -36.0, 0.0
1200.0, -48.0, 0.0
2400.0, -60.0, 0.0
4800.0, -72.0, 0.0
9600.0, -84.0, 0.0
19200.0, -96.0, 0.0
```

After the normalization file is defined, the normalization curve can be displayed by clicking the H_f button in the SigGen main menu toolbar.

Create the Segments and Components

The overall waveform in this example will consist of one segment. A frequency method, Harm, will be used to generate the segment.

The frequency-domain signal generation method, *Harm*, generates a magnitude spectrum consisting of a fundamental frequency and harmonics at specified intervals. Parameters to be specified include:

- Fundamental level in dB
- Fundamental frequency in Hz
- Frequency separation in Hz
- Roll-off in dB/component

  Roll-off controls the harmonic (component) amplitude values.
- Number of components

SigGen generates a time-domain waveform from the magnitude spectrum through use of the inverse Fast Fourier Transform (iFFT).

To create the segment

1. Select Segment from the Modify menu of the main window.
2. Set Level to 20 in order to keep the overall voltage between ± 2 volts.
3. Select a Gen Meth of Freq(8192) in order to utilize the smallest possible binwidth. See *Digital Signal Processing Applications*, Chapter 1, for more information.
4. Set Duration to 200 msec.

All other parameters remain at their default settings.

The parameters for the first segment will be:

Select:	Seg[1]
Gate Type:	Cos^2
Gate Time:	5 msec
Level:	20 dB
Gen. Meth.:	Freq(8192)
App. Meth.:	Add
Start:	0.0 msec
Duration:	200 msec

To create a component for the segment

1. Click the first button in the Components group box.
2. Enter or select the following parameters in the Component Parameters dialog box:

Call:	Harm
Fund Level:	70 dB
Fund Freq:	150 Hz
Freq Step:	150 Hz
Roll-off:	0 dB/component
No. Comp. (0 = inf):	0

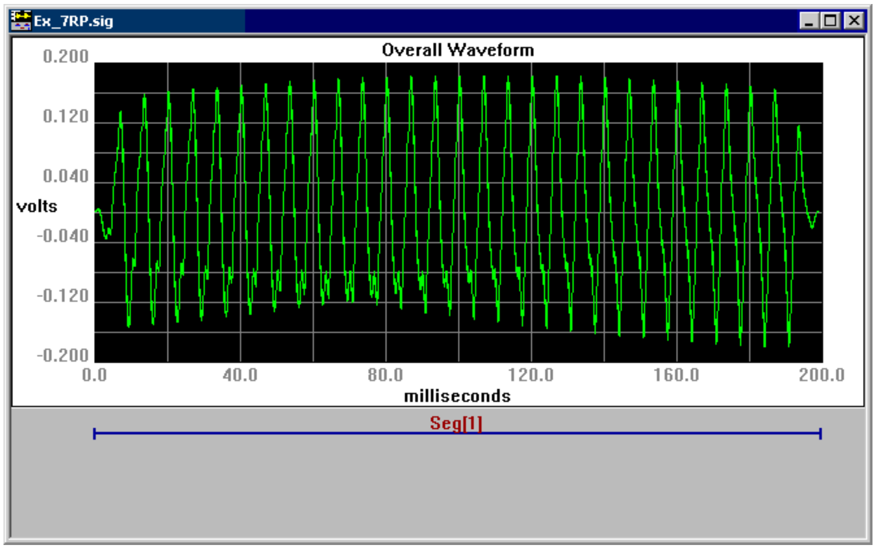
> **Note:** The Roll-off parameter remains at its default setting of 0, resulting in harmonic amplitudes equal to that of the fundamental, in this case, 90 dB. Harmonic amplitude will be controlled through the application of the normalization file. The normalization file is necessary because we desire a roll-off of 12 dB per octave. The Harm option calculates roll-off per component, not per octave, and is therefore linear rather than curvilinear.

3. Click the OK button to accept the component parameters.

The normalization file is automatically applied to the segment. This can be seen by observing the magnitude spectrum. There is a reduction of amplitude at a rate of 12 dB per octave.

To return to SigGen's main window

- ➢ Click OK in the Edit Signal Segments dialog box.

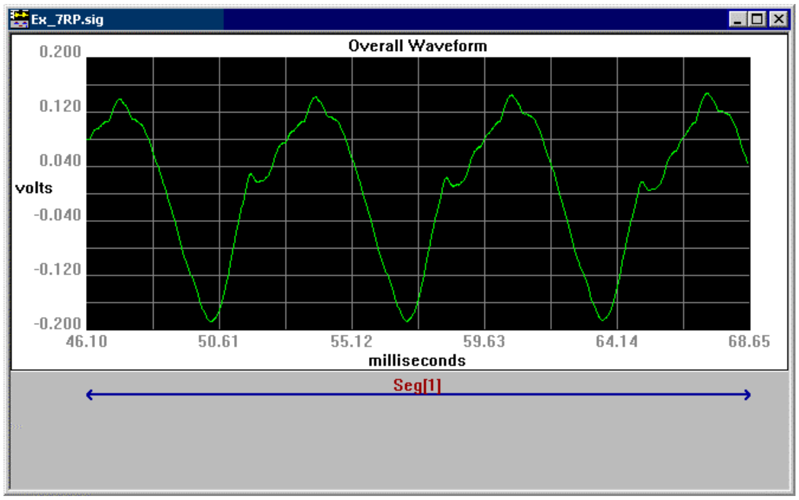


Zoom-in

The *zoom-in* feature allows you to expand a section of the waveform.

To zoom-in

1. Place the mouse at the edge of the area you wish to zoom-in.
2. Hold down the left mouse button.
3. Drag the mouse over the desired area.
4. Release the left mouse button.



Save the Signal

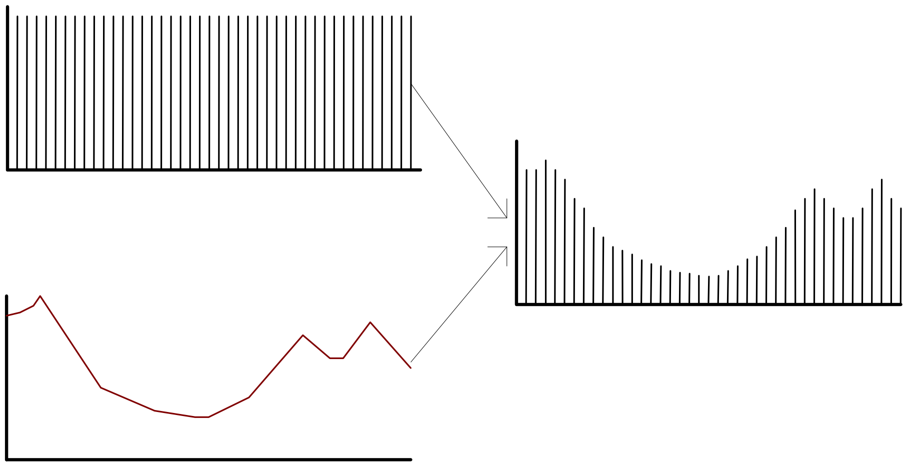
To save the signal file as a SigGen signal

1. Select Save As... from the File menu of the main window.
2. Enter the desired file name, *Ex_7*.

To save the signal file as a 16-bit binary waveform

1. Select Save Wave... from the File menu of the main window.
2. Enter the desired waveform file name, *source*.

Example 8: Producing a Steady State Vowel



- ✓ *Using a Hanning window*
- ✓ *Saving a signal for later use as an imported component*

In this example, you will generate a steady state vowel. The fundamental frequency of this vowel will be varied according to a C Major scale. You may choose to generate a steady state [ɑ], [i], or [u]. The *Harm* option will be used to generate a signal consisting of energy at the fundamental frequency and its integer multiples. Level will remain constant across harmonics. A normalization curve designed to represent the vowel spectrum will then be applied.

To generate the steady state vowel, you will need to perform the following steps:

1. Create a normalization file.
2. Run SigGen.
3. Define the signal parameters.
4. Create the frequency variable.
5. Initialize the variable.
6. Create the segment and component.
7. Play the signal.
8. Save the signal for later use as a component.

Note: Only those parameters to be entered or selected are given below. All other parameters will remain according to the default settings.

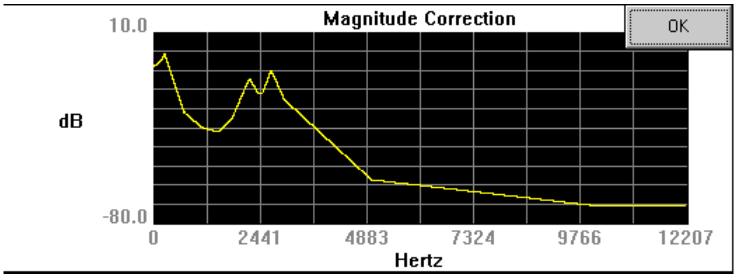
Create a Normalization File

For this example, three normalization files have already been created for your use. These file were provided with your SigGen software, and may be viewed in Notepad. They provide spectral envelopes for the following vowels:

■ [ɑ]	*EX_8_a.NRM*
■ [i]	*EX_8_i.NRM*
■ [u]	*EX_8_u.NRM*

For more information about the format of SigGen's normalization files, please refer to "Preparing to Build a Signal - The First Step" of this guide.

The normalization curve generated for [i] is presented as an example below:



Run SigGen and Create a New Signal

To run SigGen

➢ Double-click the SigGen icon

or

➢ Select SigGen from the Start\Program menu.

To create a new SigGen signal

➢ Select New from File menu, or click on the New Signal icon on the toolbar.

Configure the RPx device and RCO file

For System 3 equipment SigGenRP uses the RPx device and RCO files (See RPvdsEx help) to play out signals.

To configure the RPx device and select the RCO file

1. Select RP devices|Device A from the Modify menu or click on the “A” icon on the toolbar.
2. Select from the Standard file menu “Sweep Play, 2 Chan 50K (max 100K)”.
3. Select RP2 Processor as device type

4. Select Index 1.
5. Deselect “Use sample rate specified in RCO file”
6. Under Bandwidth and Timing change the Standard Sample Rate to 25kHz. (Ignore the Number of Time slices).

Define the Signal Parameters

To define the signal parameters

1. Select Signal from the Modify menu of the main window.
2. Enter or select the following parameters in the Signal Parameters dialog box:

Name:	*Ex_8*
Duration:	1000 milliseconds
Calibration	
Level:	65 dB
Use Norm File	☒
File Name	*C:\TDT\SigGenRP\Examples\EX_8_a.NRM* *or* *C:\TDT\ SigGenRP\Examples\EX_8_i.NRM* *or* *C:\TDT\ SigGenRP\Examples\EX_8_u.NRM*

In this example, the signal will be low-pass filtered such that the cut-off frequency is at or below 10,000 Hz.. Therefore, we have increased the sample period from the default of 20 µ seconds to 50 µseconds.

After the normalization file is defined, the normalization curve can be displayed by clicking the H_f button in the SigGen main menu toolbar.

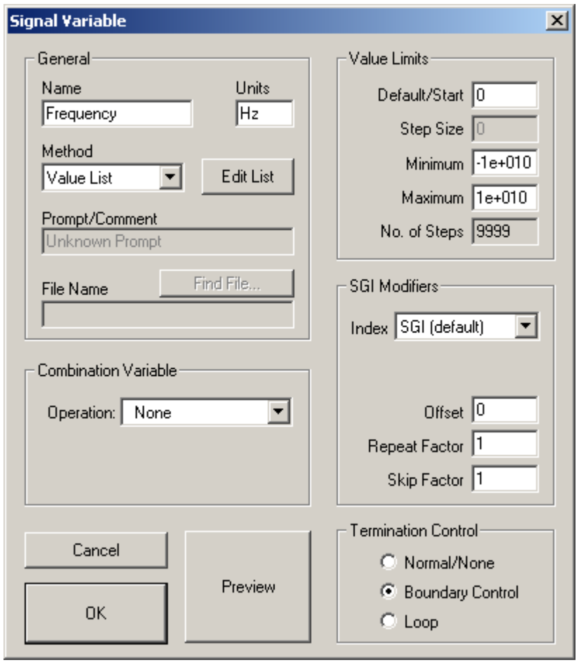
Create the Frequency Variable

Values for fundamental frequency and harmonic interval will be varied according to a C Major scale using equal tempered tuning.

To create the variable, Frequency

1. Select Signal from the Modify menu of the main window.
2. Click the Edit button in the Variables group box of the Signal Parameters dialog box.
3. Enter or select the following parameters in the Signal Variable dialog box:

Name:	*Frequency*
Method:	Value List



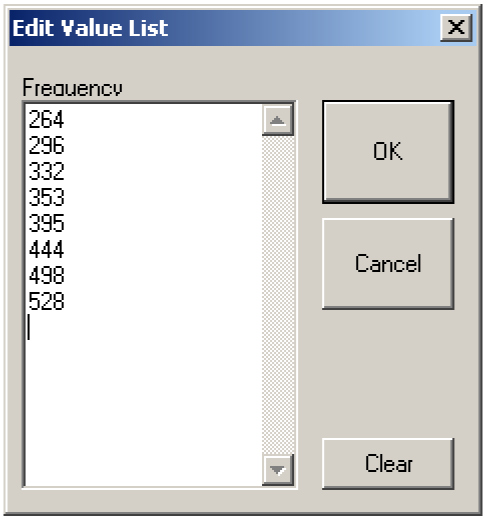
Name: Frequency
Units: Hz
Method: Value List
Prompt/Comment: Unknown Prompt
File Name:
Operation: None
Default/Start: 0
Step Size: 0
Minimum: -1e+010
Maximum: 1e+010
No. of Steps: 9999
Index: SGI (default)
Offset: 0
Repeat Factor: 1
Skip Factor: 1
○ Normal/None
● Boundary Control
○ Loop

After selecting the Value List method you must then edit the list. Click on the Edit List button.

Once the empty Value List window pops add the specific frequencies.

```
264
296
332
353
395
444
498
528
```

Note: You can simply copy and paste the frequencies listed above directly into the empty Value List.


Edit Value List

Frequency

264
296
332
353
395
444
498
528

To accept the Value List Changes and return to the Signal Variables dialog box

➢ Click the OK button in the Edit Value List dialog box.

To accept the variable parameters and return to the Signal Parameters dialog box

➢ Click the OK button in the Signal Variable dialog box.

To accept the signal parameters and return to the SigGen main window

➢ Click the OK button in the Signal Parameters dialog box.

Create the Segments and Components

The overall waveform in this example will consist of one segment. A frequency method, Harm, will be used to generate the segment.

The frequency-domain signal generation method, Harm, generates a magnitude spectrum consisting of a fundamental frequency and harmonics at specified intervals. Parameters to be specified include:

- Fundamental level in dB
- Fundamental frequency in Hz
- Frequency separation in Hz
- Roll-off in dB/component
- Number of components

SigGen generates a time-domain waveform from the magnitude spectrum through use of the inverse Fast Fourier Transform (iFFT).

In this example, you will be employing a Gate Type of Hann. Hann specifies the application of a *Hanning window*. A Hanning window does not require specification of Gate Time. Changing the value of the Gate Time parameter will have no effect on the Hanning window.

To create the segment

1. Select Segment from the Modify menu of the main window.
2. Select a Gate Type of Hann.
3. Set Level to 0 in order to keep the overall voltage between ± 1 volt.
4. Select a Gen Meth of Freq(8192) in order to utilize the smallest possible binwidth. See *Digital Signal Processing Applications*, Chapter 1, for more information on binwidth.
5. Set Duration to 1000 msec.

All other parameters remain at their default settings.

The parameters for the first segment will be:

Select:	Seg[1]
Gate Type:	Hann
Gate Time:	5 msec
Level:	0 dB
Gen. Meth.:	Freq(8192)
App. Meth.:	Add
Start:	0.0 msec
Duration:	1000 msec

To create a component for the segment

1. Click the first button in the Components group box.
2. Enter or select the following parameters in the Component Parameters dialog box:

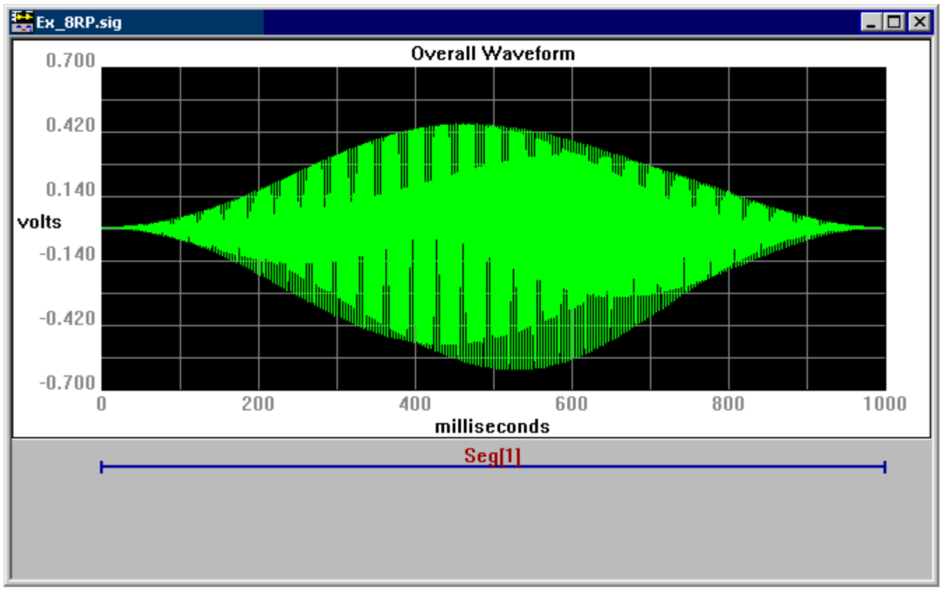
Call:	Harm
Fund Level:	70 dB
Fund Freq:	Choose the variable, *Frequency*
Freq Sep:	Choose the variable, *Frequency*
Roll-off:	0 dB/component
No. Comp. (0=inf.):	0

3. Click the OK button to accept the component parameters.

The normalization file is automatically applied to the segment. This can be seen by observing the magnitude spectrum.

To return to SigGen's main window

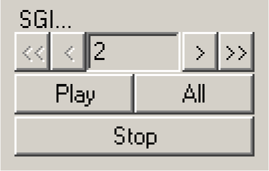
➢ Click OK in the Edit Signal Segments dialog box.



Play the Signal

The variable file specifies eight frequency values that correspond to a C major scale originating at middle C. Any note in the scale may be played by selecting the appropriate SigGen Index (SGI) value.

Note	SGI
Middle C	1
D	2
E	3
F	4
G	5
A	6
B	7
C	8



To play a specific note in the C Major scale

1. Click < or > until the appropriate SGI is displayed in the SGI field.
2. Click the Play button.

To play the entire C Major scale

➢ Click the All button.

To stop play

- Click the Stop button.

Save the Signal for Later Use As a Component

Components may be created from any 16-bit binary waveform or floating point waveform. It is therefore possible to save any SigGen signal as a 16-bit integer waveform and later use it as a *component* in a multi-component signal.

Waveforms that will be used later as components **must** be saved with the extension, *.000*. These waveforms may be saved as 16-bit integer waveforms by using the Save Wave... command found in the File Menu.

Note: When saving SigGen signals as 16-bit integer waveforms, it is important to keep in mind that only the current waveform will be saved. No variable information is saved. Before saving, make sure the current waveform reflects the desired variable values by setting the SGI to the correct value. In our example, if you wish to save a vowel with a pitch of G, you would set the SGI to 5, then save the waveform.

To save the vowel for later use as a component

1. Set the SGI index to the value that produces the vowel at the desired frequency.
2. Choose Save Wave... from the File menu.
3. Type in the desired file name followed by the mandatory *.000* extension.

Chapter 10 Examples: Auditory Sciences

In this chapter, examples of auditory stimuli are presented. Each example introduces new SigGen concepts. The examples found in this chapter are listed below. Also listed are the concepts introduced in each example.

Example 9: Amplitude Modulated (AM) Noise

- Calibration of dB in SPL
- DC Offset

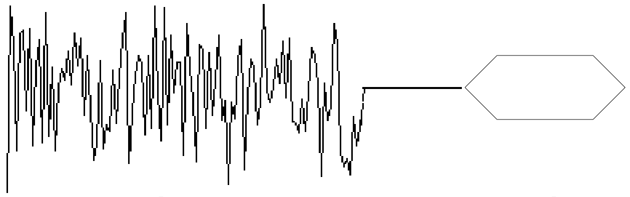
Example 10: Forward Masking

- Calibration of dB in SL (Correction factors)
- Combination Variables
- Dynamic Variables

Example 9: Amplitude Modulated (AM) Noise

Example to be included in a future version of this document.

Example 10: Forward Masking



In this example, we will create a forward masking experiment. The masker will consist of 150 ms of band-limited Gaussian noise presented at a Sensation Level (SL) of 65 dB. The probe will consist of a 40 ms, 1000 Hz pure tone. The interval between the masker and probe, ΔT, will vary from 10 ms to 100 ms in 10 ms increments. For each value of ΔT, the subject's forward masking threshold will be determined by increasing the probe intensity level from 0 dB SL in 5 dB increments until the probe signal can be detected 2 out of 3 times or until the probe level reaches 70 dB SL.

Concepts introduced:

- ✓ *Calibration in SPL*
- ✓ *Calculation of SL Correction Factors*
- ✓ *Combination Variables*
- ✓ *Dynamic Variables*

Several new concepts are introduced. First, we will calibrate SigGen so that all signal level measures are reported in dB SPL as measured at the system output. Because we want to set the masker and probe levels based on the subject's Sensation Level, we will also need to calculate correction factors for the masker and the probe. For each value of ΔT, probe intensity will be varied. This type of nested variable manipulation requires the use of a Variable File.

The new concepts described above are also listed in the adjacent column, and will be marked by the SigGen icon when they are introduced.

To generate this signal, you will need to perform the following steps:

1. Run SigGen.
2. Obtain dB SPL reference value.
3. Calculate SL correction factors.
4. Define the signal parameters.
5. Create a variable file.
6. Define the variables for ΔT and probe intensity.
7. Define a dynamic variable which is also a combination variable.
8. Create the noise and probe tone segments.

Note: Only those parameters to be entered or selected are given below. All other parameters will remain according to the default settings.

Run SigGen and Create a New Signal

To run SigGen

- Double-click the SigGen icon

  or

- Select SigGen from the Start\Program menu.

To create a new SigGen signal

- Select New from File menu, or click on the New Signal icon on the toolbar.

Obtain dB SPL Reference Value

In order to calibrate SigGen in dB SPL, you must have some means for determining the output SPL of the system. The process is as follows:

1. Open a new signal window.
2. Set duration to 2000 ms and calibrate SigGen so that 1 dB equals 1 volt.
3. Build a pure tone signal of 1000 Hz with a an amplitude of 1 dB (peak amplitude of 1 volt).
4. Play the signal and measure the output SPL. The measured output becomes your dB reference.

To open a new signal window

- Choose New from the File menu.

To set duration and calibrate 1 dB = 1 volt

1. Choose Signal... from the Modify menu.

   You will see the Signal Parameters dialog box.
2. Enter Tone in the Name Field.
3. Enter 2000 ms in the Duration field.
4. Enter 1 dB in the Calibration Level field.
5. Click OK to return to the main signal window.

To build a 1 dB, 1000 Hz pure tone

1. Choose Segment... from the Modify menu.

   You will see the Edit Signal Segments dialog box.
2. Entire 2000 ms in the Duration field.
3. Click the first Component button.

4. Set the following Component parameters:

Call:	Tone
Level:	1 dB
Frequency:	1000 Hz
Phase:	0 Degrees

5. Click OK to accept the Component parameters.
6. Click OK to accept the Segment parameters.

To play the calibration tone and measure the output

1. Click the play button.
2. Measure the output SPL during the middle portion of the signal.

   For our example, we will assume this measured output SPL was 85 dB.

Calculate SL Correction Factors

In order to calculate correction factors for our masker and probe, we must create a separate calibration signal for each. We will use a modified version of our existing signal, Tone, to determine the SL correction factor for the 1000 Hz probe. We will then create a noise signal, Masker. This will be used to determine the SL correction factor for the masker.

Probe SL Correction Factor

In order to determine the probe SL correction factor, we will present the probe to the subject while increasing the intensity from 0 dB SPL in 5 dB steps until the tone is barely perceived. The probe will be repeated several times at each intensity level depending on the experiment. The process consists of the following steps:

1. Calibrate the signal Tone in dB SPL.
2. Define the intensity variable, Probe_Level.
3. Modify the segment parameters and assign the variable, Probe_Level
4. Present the signal.
5. Record the SPL level.

To calibrate Tone in dB SPL

1. Choose Signal... from the Modify menu.

   You will see the Signal Parameters dialog box.
2. Enter 85 dB in the Calibration Level field.

   Recall that 85 dB is our SPL calibration reference for one volt.

To define the intensity variable, Probe_Level

1. Click the Edit button in the Variables group box of the Signal Parameters dialog box.
2. Enter the following parameters.

Name:	*Probe_Level*
Method:	Linear Step
Default/Start:	5
Step Size:	5

To accept the parameters

1. Click the OK button in the Signal Variable dialog box.
2. Click OK to return to the main signal window.

To modify the segment and assign the variable, Probe_Level

1. Choose Segment... from the Modify menu.

   You will see the Edit Signal Segments dialog box.
2. Entire 40 ms in the Duration field.
3. Click the first Component button.
4. Set the following Component parameters:

Call:	Tone
Level:	Probe_Level
Frequency:	1000 Hz
Phase:	0 Degrees

5. Click OK to accept the Component parameters.
6. Click OK to accept the Segment parameters.

To present the signal

1. Prepare the subject for signal presentation.
2. Instruct the subject to signal when they hear the tone.
3. Click the All button.
4. Click Stop when the subject signals that the tone has been perceived.

To record the probe SL correction factor

➢ Note the current value of Probe_Level as shown in the Variables box.

This will be the probe SL correction factor. The probe SL correction factor is a dB SPL number that corresponds to 0 SL for the subject. For this example, we will say the probe correction factor equals 35 dB SPL.

Masker SL Correction Factor

We will determine the masker SL correction factor in much the same way as we did the probe SL correction factor. The process consists of the following steps:

1. Open a new signal window.
2. Calibrate the signal Tone in dB SPL.
3. Define the intensity variable, Masker_Level.
4. Modify the segment parameters and assign the variable, Masker_Level
5. Present the signal.
6. Record the SPL level.

To open a new signal window

➢ Choose New from the File menu.

To calibrate Tone in dB SPL

1. Choose Signal... from the Modify menu.

   You will see the Signal Parameters dialog box.
2. Enter Masker in the Name field.
3. Enter 300 ms in the Duration field.
4. Enter 85 dB in the Calibration Level field.

   Recall that 85 dB is our SPL calibration reference for one volt.

To define the intensity variable, Masker_Level

1. Click the Edit button in the Variables group box of the Signal Parameters dialog box.
2. Enter the following parameters.

Name:	*Masker_Level*
Method:	Linear Step
Default/Start:	5
Step Size:	5

To accept the parameters

1. Click the OK button in the Signal Variable dialog box.
2. Click OK to return to the main signal window.

To modify the segment and assign the variable, Masker_Level

1. Choose Segment... from the Modify menu.

   You will see the Edit Signal Segments dialog box.
2. Enter 150 ms in the Duration field.
3. Click the first Component button.

4. Set the following Component parameters:

Call:	Gauss
Level:	Masker_Level

5. Click OK to accept the Component parameters.
6. Click OK to accept the Segment parameters.

To present the signal

1. Prepare the subject for signal presentation.
2. Instruct the subject to signal when they hear the tone.
3. Click the All button.
4. Click Stop when the subject signals that the tone has been perceived.

To record the probe SL correction factor

- Note the current value of Masker_Level as shown in the Variables box.

  This will be the masker SL correction factor. The masker SL correction factor is the number of dB SPL that corresponds to 0 SL for this subject. For this example, we will say the masker correction factor equals 30 dB SPL.

Configure the RPx device and RCO file

For System 3 equipment SigGenRP uses the RPx device and RCO files (see RPvdsEx help) to play out signals. The

To configure the RPx device and select the RCO file

1. Select RP devices|Device A from the Modify menu or click on the “A” icon on the toolbar.
2. Select from the Standard file menu “Sweep Play, 2 Chan 50K (max 100K)”.
3. Select RP2 Processor as device type
4. Select Index 1.
5. Deselect “Use sample rate specified in RCO file”
6. Under Bandwidth and Timing change the Standard Sample Rate to 25kHz. (Ignore the Number of Time slices).

Define the Signal Parameters

We are now ready to construct our masking experiment. Although we could build the signal by modifying one of our calibration signals, it will be easier to describe the process if we simply start from scratch.

Previously, we determined the output of the system based on a signal of one volt peak amplitude. For purposes of this example, we said the output was 85 dB SPL. We can now calibrate SigGen so that all level values represent output SPL. Calibration is accomplished during the signal parameter definition process.

To define the signal parameters

1. Close all existing signal windows.
2. Choose New from the File menu.
3. Select Signal from the Modify menu of the main window.
4. Enter the following parameters in the Signal Parameters dialog box:

Name:	*Ex_10*
Duration:	300 msec
<u>Calibration</u>	
Level:	85 dB

To accept the parameters and return to the main window

➢ Click the OK button.

Define the Variables

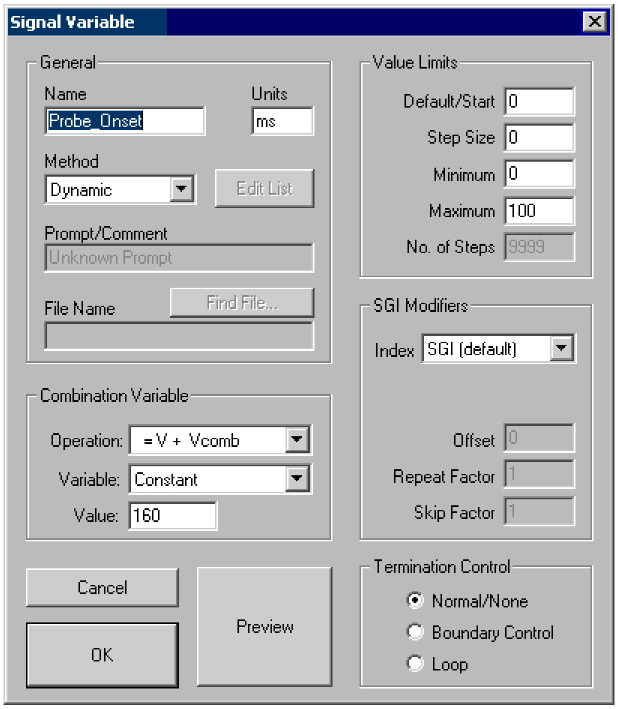
Each variable must be defined in SigGen.

In a forward masking experiment, Probe_Onset variable will be varied according to the response of the subject. A Dynamic variable will allow an application software to control the value of the variable.

Probe_Onset is defined as the delay from the masker to illustrate its physical meaning. While generating stimulus, this delay must be in reference to the Beginning of the signal. A Combination variable is used below to implement this scale conversion (adding 160 ms).

To define the variable for probe onset

1. Select Signal from the Modify menu of the main window.
2. Click the Edit button in the Variables group box of the Signal Parameters dialog box.
3. Enter the parameters seen below:



Name: Probe_Onset
Units: ms
Method: Dynamic
Prompt/Comment: Unknown Prompt
Default/Start: 0
Step Size: 0
Minimum: 0
Maximum: 100
No. of Steps: 9999
Index: SGI (default)
Offset: 0
Repeat Factor: 1
Skip Factor: 1
Operation: = V + Vcomb
Variable: Constant
Value: 160
Termination Control: ● Normal/None ○ Boundary Control ○ Loop

To accept the parameters and return to the Signal Parameters dialog box

➢ Click the OK button in the Signal Variable dialog box.

To define the variable for probe level

1. Select the next available variable (........) in the Variables group box of the Signal Parameters dialog box.
2. Click the Edit button.
3. Enter or select the following parameters in the Signal Variable dialog box:

Name:	*Probe_Level*
Method:	Linear Step
Start/Default:	5
Step Size:	5
Minimum:	5
Maximum:	70
Termination Control:	Boundary Control

4. Click OK to accept the variable parameters.
5. Click OK to return to the main window.

Create the Segments and Components

This signal will consist of two segments, the masker and the probe. The masker will consist of a 150 ms Gaussian noise presented at 65 dB SL. The probe will be a 1000 Hz pure tone presented at varying intensity levels. The interval between the masker and the probe will be increased by varying the probe onset time.

To create the masker segment

1. Select Segment from the Modify menu of the main window.
2. Enter or select the following parameters in the Edit Signal Segments dialog box:

The parameters for the first segment will be:

Select:	Seg[1]	
Gate Type:	Cos2	
Gate Time:	5 msec	
Level:	30 dB	⇐ *Conversion factor for masker*
Gen. Meth.:	Time	
App. Meth.:	Add	
Start:	0.0 msec	
Duration:	150 msec	

Note: The Level parameter above has been set to the masker SL conversion factor, 30 dB. In a future section, we will define the intensity of the masker section as 65 dB SL. Since SigGen is calibrated in SPL, it will interpret the masker intensity as 65 dB SPL. In order to correct for this, we must increase the overall segment level by 30 dB, yielding an SPL of 95 dB SPL. This intensity is the equivalent of 65 dB SL for this subject.

To create the component for the masker segment

1. Click the first button in the Components group box.
2. Enter or select the following parameters in the Component Parameters dialog box:

Call:	Gauss
Level:	65 dB

To accept the parameters and return to the Edit Signal Segments dialog box

➢ Click the OK button.

To create the probe segment and components

1. Click the New button in the Edit Signal Segments dialog box.
2. Create the segments and components according to the following parameters.

Probe segment parameters:

Select:	Seg[2]	
Gate Type:	Cos2	
Gate Time:	5 msec	
Level:	35 dB	⇐ *Conversion factor for probe*
Gen. Meth.:	Time	
App. Meth.:	Add	
Start:	Probe_Onset	
Duration:	40 msec	

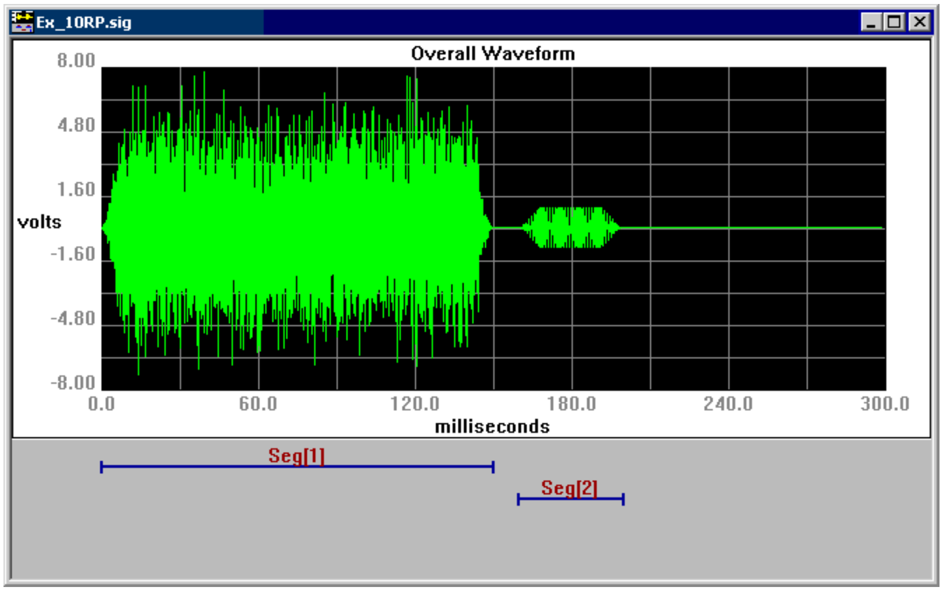
Probe segment component:

Call:	Tone
Level:	Probe_Level
Frequency:	1000 Hz
Phase:	0 Degrees

Note: The Level parameter above has been set to the probe SL conversion factor, 35 dB. The probe Segment Level parameter, Probe_Level is defined in dB SL. Since SigGen is calibrated in SPL, it will incorrectly interpret these intensity values. By increasing the overall segment level by 35 dB, an SPL level that corresponds to the desired SL level is produced.

To accept the segment parameters and return to the main window

➢ Click the OK button.



Part 3 Appendices

Appendix A Menus and Toolbars

The File Menu

The File Menu provides a standard Windows method for creating, saving, opening, closing, and printing files.

New... Opens a new Signal Window.

Open... Opens and displays an existing signal file.

Close Closes the current signal window.

Save and **Save As...** These menu options save signals in SigGenRP's own format. Files saved in this format use the default extension *.sig*.

Save Wave... Saves a waveform as a 16-bit integer binary file. Files saved in this format use the default extension *.16*.

Print... Prints the current waveform.

Print Preview Allows you to preview the waveform as it will be printed.

Print Setup... Allows you to select a printer, define the paper size, and choose a page orientation.

The Modify Menu

The Modify Menu provides commands that allow you to specify and edit signal and segment parameters, configure peripheral devices, and modify SigGenRP display.

Signal... Opens the Signal Parameters dialog box, from which you can define variables and specify parameters that will effect the entire signal.

Segment... Opens the Edit Signal Segments dialog box, from which you can specify the segment parameters and design the components of the segment.

Peripherals... Opens the Peripheral Configuration dialog box, from which you can configure TDT's programmable hardware (e.g., attenuators, waveform generators, switches and programmable filters).

Note: Not available unless using System II.

RP Devices ► Opens the Setup RP Device dialog box, from which you can configure TDT’s Real-time processors.

Display... Opens the Display Setup dialog box, from which you can specify colors for different display areas and select cursor types.

The View Menu

From the View Menu, you can choose to display or hide the Toolbar, Status Bar and Cursor Box. If a Normalization Curve is used to correct the signal, you can display the Normalization Curve.

The Operations Menu

The Operations menu provides commands that calculate the signal, set the value of signal variables, and control signal preview.

Calculate Causes the signal displayed in the current signal window to be recalculated based on the current SGI and parameter settings. This operation will update the waveform displayed in the Signal Window.

Initialize Variables Causes the SigGen Index (SGI) to be reset to 1, restoring all signal variables to their starting values. Recalculates and updates the waveform displayed in the Signal Window.

Prompt Level #1 When a Prompt Level #1 variable is defined, this operation will display the Input Request dialog box and prompt you for the value of the variable. This box will be repeatedly displayed if there are more than one Prompt Level #1 variable defined. This box will not be displayed if there is no Prompt Level #1 variable defined.

Prompt Level #2 When a Prompt Level #2 variable is defined, this operation will display the Input Request dialog box and prompt you for the value of the variable. This box will be repeatedly displayed if there are more than one Prompt Level #2 variable defined. This box will not be displayed if there is no Prompt Level #2 variable defined.

Note: When there are more than one Prompt variable defined, values entered to Input Request box will be assigned to the variables according to their order in the variable table.

Increment SGI Increments the SigGen Index by 1. This operation will recalculate and update the current waveform.

Advance SGI by 10 Increments the SigGen Index by 10. This operation will recalculate and update the current waveform.

Decrement SGI Decrements the SigGen Index by 1. This operation will recalculate and update the current waveform.

Decrease SGI by 10 Decrements the SigGen Index by 10. This operation will recalculate and update the current waveform.

Play Waveform Plays the signal at the current SigGen Index.

Play All Plays the signal for all SigGen Indices beginning at SGI = 1. If there is no boundary control for SGI through a SigGen variable, signal will play until it is stopped manually.

Stop Playing Halts signal play.

The Window Menu

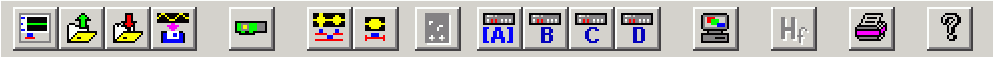
SigGenRP supports the Multiple Document Interface (MDI) standard. The Window menu provides standard Windows commands that allow you to easily manipulate multiple signal windows.

The Help Menu

From the Help Menu, you can display the About SigGen message box or access the SigGen help.

The Toolbar

You can find quick point-and-click access to the most common commands on the SigGenRP Toolbar.



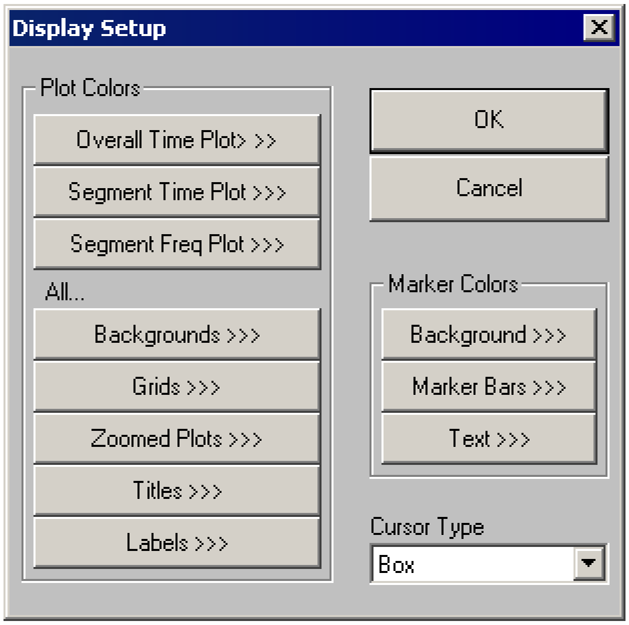
Click		To
	New	Open a new Signal Window
	Open	Open an existing SigGen signal file
	Save As	Save a SigGen signal file
	Save Wave	Save to a 16-bit binary file
	Calculate	Calculate the current signal
	Modify Signal	Edit signal parameters
	Modify Segment	Edit segment parameters
	Configure Peripherals	Configure programmable TDT hardware.
	Configure Device A	Configure programmable Real-time processor.
	Configure Device B	Configure programmable Real-time processor.
	Configure Device C	Configure programmable Real-time processor.
	Configure Device D	Configure programmable Real-time processor.
	Display Setup	Customize display colors and cursor types
	Norm Curve	View Normalization Curve
	Print Preview	Preview the waveform as it will be printed
	SigGen Help	Access SigGen Help

Appendix B Customizing Your Display

You may find it useful to alter the default SigGenRP screen configuration. Many of the display settings (such as window title bar color and size, box border color and size, etc.) can only be altered from Windows Control Panel. Customizing these system settings is beyond the scope of this user's guide, and you may read your Windows User's Manual provided with your Windows software. Display settings that can be altered in SigGenRP can be customized from the Display Setup dialog box.

To access the Display Setup dialog box

➢ Choose Display... from the Modify menu.



Cursor Type
Box

From the Display Setup dialog box you may configure the following:

- Colors
- Cursor Type

Customizing Colors

You can customize the color of numerous screen and plot elements.

Plot Colors

You may customize the color of waveform plots:

- The Overall Time Plot

  The Overall Time Plot is the plot of the completed signal. This plot is displayed in the Signal Window of SigGenRP's main window. This is the color of the waveform.

- The Segment Time Plot

  The Segment Time Plot displays the time-domain waveform of an individual segment. This plot is displayed in the Time Form display of the Edit Signal Segments window.

- The Segment Frequency Plot

  The Segment Time Plot displays the frequency-domain waveform of an individual segment. This plot is displayed in the Magnitude Spectrum display of the Edit Signal Segments window.

You may also customize the color of other plot features. The changes made here will apply to both the signal window and segment window:

- Background

  Background color is the plot background color of both signal and segment windows.

- Grids

  Grids color is the plot grid color of both signal and segment windows

- Zoomed Plots

  Zoomed Plots color is the waveform color in both signal and segment windows when the plot is zoomed in.

- Titles

  Title color is the color of plot title and x-axis and y-axis titles of both signal and segment windows

- Labels

  Label color is the color of x-axis and y-axis scale numerical labels of both signal and segment windows

Marker Colors

The Segments Area of the Signal Window displays graphic representations of the durations and temporal positions of each segment in the signal. These graphic representations are known as *markers*.

You may customize the color of the following:

- Marker Background

The Marker Background is the background of the Segments Area.

- Marker Bars

  Marker Bars are the graphic representations of the durations and temporal positions of each segment.

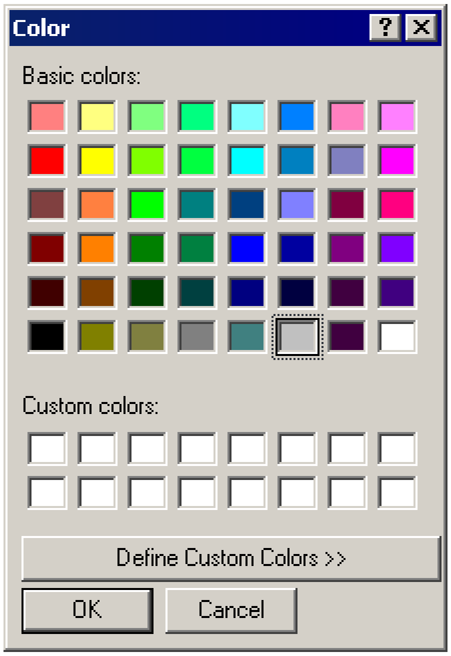
- Marker Text

  Marker Text is the text label of the marker.

To access the Color dialog box

➢ Click the appropriate Color button of the Display Setup box.

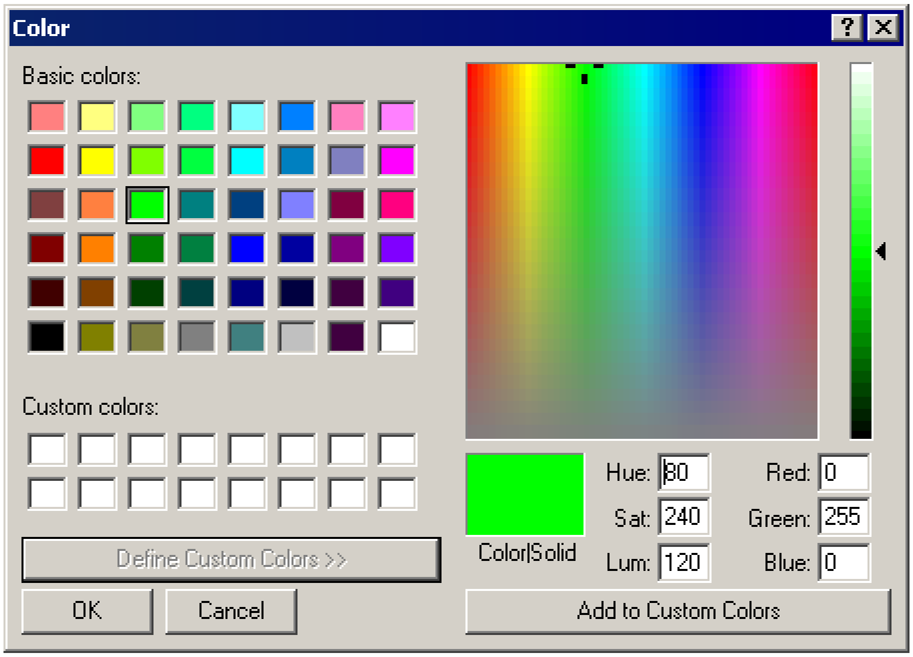
From the Color dialog box, you may select one of the available colors or design your own custom color.



To design custom colors

➢ Click the Define Custom Colors button.

The Color dialog box will expand to provide a color palette from which you can design your custom colors by selecting a point in the palette. You can also design your color by using one of the two color models (see below) provided in SigGen. Sample color is displayed in both actual display color and solid color. Solid color is a closest representation of the selected color with basic system color. Clicking at the Add to Custom Colors will add the selected color to one of the Custom colors. Clicking at a specific color button of the Custom colors and then click Add to Custom Colors button will add the color to the specified button.



Hue: 80
Sat: 240
Lum: 120
Red: 0
Green: 255
Blue: 0

Color Models

SigGenRP provides two different color models for creating display colors: RGB (red, green, blue) and HSB (hue, saturation, brightness). You can create colors using the model you are most comfortable with.

RGB Model

The RGB color model uses percentages of red, green and blue to create colors. Each component has 255 levels of intensity, ranging from black to the component's full intensity. Thus, to produce pure red for example, set Red to 255 and Green and Blue to 0. Similarly, set Green to 255 and the others to 0 to produce pure green. To produce a dark but pure shade of one of the three, lower its setting while leaving the others at 0.

White is produced by setting all three components to 255. Setting them all to 0 produces black. Equal amounts of each produces varying shades of gray.

HSB Model

The HSB model, creates color by varying three parameters: hue, saturation and brightness (lumens). Hue refers to the quality which makes a particular color different from another. Blue, red, and green, for example, are all hues. Saturation refers to the purity or intensity of a color. By varying the intensity, you can make the color lighter or darker. Brightness refers to the percentage of black in a color, where 0 percent is black and 255 percent is white.

Customizing the Cursor Type

You may place a cursor in any of the SigGenRP plots. Cursors can be of the following types:

- Box
- Small Box
- Large Box
- Vertical Line
- Fat Vertical Line
- Small 'X'
- Fat 'X'
- Cross Hair

To customize a Cursor Type

➢ Select the desired cursor from Cursor Type box.

Appendix C Using SigGenRP with System II

This appendix provides the information needed to use SigGenRP with System II Hardware.

While SigGenRP continues to support System II hardware we recommend continuing to use the version of SigGen that was purchased with System II hardware whenever possible. SigGenRP is configured to for System 3 by default. If you decide to use it with System II, contact TDT for assistance with reconfiguring the software.

Signal Setup

Signal parameters are defined from the Signal Parameters dialog box, which can be accessed from Modify menu, or from the Modify Signal icon on the toolbar. The signal parameters are defined in the same way as described for System 3 with the exceptions described below.

To access the Signal Parameters dialog box

- ➢ Click Signal... from the Modify menu, or click on the Modify Signal icon on the toolbar.

Use Sys 3

The check box should be cleared.

Sample Period

Sample period indicates the length of time in microseconds between successive data points in the time-domain waveform.

When other factors that affect signal generation are held constant, the value of the sample period affects signal generation in several ways:

Change in Sample Period	**Resulting Maximum Frequency**	**Resulting Binwidth**
Decrease	Increase	Decrease
Increase	Decrease	Increase

To determine the Sample Period

1. Determine the *maximum frequency* (f_{max}).

   f_{max} will be the lower of the following:

   - The frequency (in Hz) of the highest component desired in the signal.

- The corner frequency (in Hz) of the filter being used during signal presentation.

2. Determine a *sampling frequency* (in Hz) that is at least 2 or 2.5 times f_{max} to avoid aliasing.

   Theoretically *sampling frequency* must be at least 2 times f_{max} to avoid aliasing. In the real world, if f_{max} is at or near the filter's corner frequency, the cut-off slope of the filter must be considered. To avoid aliasing with TDT's FT5 filter, multiply the f_{max} by 2.5.

   $$\text{sampling frequency} = 2.5 \cdot f_{max}$$

   For a more accurate representation of a signal, you may need to select a higher sampling frequency. For example, if you want to present a square waveform, a sampling frequency selected by the above formula will produce a saw-type waveform, and sinusoidal-like waveform after an anti-aliasing filter. In such a case, sampling frequency must be increased based on your accuracy demand.

3. Compute the inverse of the *sampling frequency* (in Hertz) to determine the *Sample Period* (in seconds)**.**

   Sample Period (seconds) = 1/sampling frequency (Hertz)

4. Convert Sample Period in seconds to microseconds.

   Sample Period (microseconds) = Sample Period (seconds) $\cdot 10^6$

Onset Delay

Onset delay is controlled through use of a TG6. During signal play, Channel 2 of the TG6 is programmed such that it's output is delayed *n* milliseconds, where *n* is the value defined in the O/S Delay field.

Note: In order to delay signal presentation through use of the TG6, the playback D/A channel must be triggered through Channel 2 of the TG6.

Configuring Peripherals

SigGenRP allows you to configure various programmable TDT devices. These device configurations may be static or may vary across SGIs, and can be used in application programs.

- This feature is only available with System II hardware.

The programmable devices which can be configured in SigGenRP include:

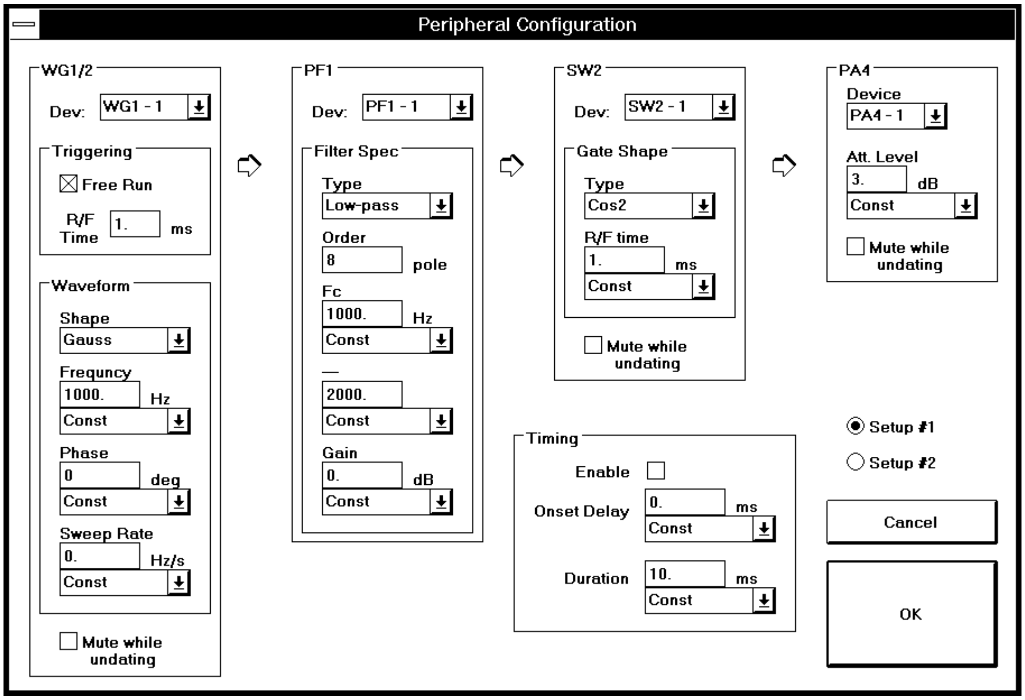
- Waveform generators: WG1s and WG2s
- Programmable filters: PF1s
- Cosine switches: SW2s
- Programmable attenuators: PA4s

Only one of each group of above devices can be configured in one signal. For example, you cannot control two PA4s at the same time. You cannot configure a device that you do not have in your hardware system, even SigGenRP lists it in the Peripheral Configuration dialog box. Peripherals may be configured from the Peripheral Configuration dialog box.

To access the Peripheral Configuration dialog box

- ➢ Choose Peripherals... from the Modify menu, or

  Click on the Peripheral Configuration icon on the toolbar.



Peripheral Configuration

WG1/2
Dev: WG1 - 1
Triggering
☒ Free Run
R/F Time 1. ms
Waveform
Shape Gauss
Frequncy 1000. Hz
Const
Phase 0 deg
Const
Sweep Rate 0. Hz/s
Const
□ Mute while undating

PF1
Dev: PF1 - 1
Filter Spec
Type Low-pass
Order 8 pole
Fc 1000. Hz
Const
—
2000.
Const
Gain 0. dB
Const

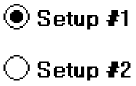
SW2
Dev: SW2 - 1
Gate Shape
Type Cos2
R/F time 1. ms
Const
□ Mute while undating

PA4
Device PA4 - 1
Att. Level 3. dB
Const
□ Mute while undating

Timing
Enable □
Onset Delay 0. ms
Const
Duration 10. ms
Const

● Setup #1
○ Setup #2

Setups



● Setup #1
○ Setup #2

Peripheral configuration parameters may be saved as "Setups." Setups provide a quick an easy means for changing peripheral configurations. You may save up to two setups.

To save a peripheral configuration setup

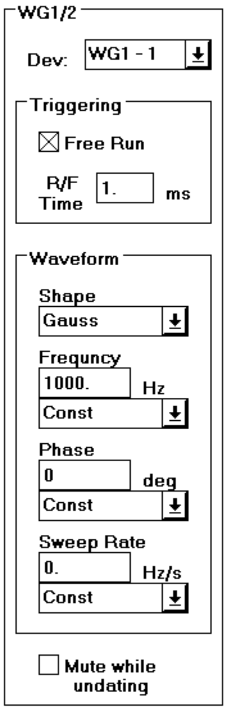
1. Specify peripheral configuration parameters.
2. Click Setup #1 or Setup #2.
3. Click OK.

To quickly change between setups

- Click the desired setup.

WG1/2

SigGenRP supports two TDT waveform generators, the WG1 and WG2. You may configure a WG1 or WG2 by specifying the parameters found within the WG1/2 group box.



WG1/2

Dev: WG1 - 1

Triggering

☒ Free Run

R/F Time: 1. ms

Waveform

Shape: Gauss

Frequncy: 1000. Hz, Const

Phase: 0 deg, Const

Sweep Rate: 0. Hz/s, Const

□ Mute while undating

Dev.

There may (or may not) be more than one WG1 or WG2 in your system. You must specify which WG1 or WG2 you wish to configure. This is accomplished by choosing the desired device from the Dev. list.

Triggering

Free Run. Checking Free Run causes the WG1 or WG2 to generate noise continuously. During Free Run, triggering is ignored. If you wish to initiate waveform generation through use of a trigger, do not check Free Run.

R/F Time. Waveform output generated by the WG1 or WG2 can be gated on and off by a linear gating feature. This field specifies the gate's rise and fall time after triggering. You cannot use a variable for this value.

Waveform

Shape. This parameter allows you to specify the waveform shape. You may choose any one from the three available WG1/2 waveform shapes:

■	Gauss	Gaussian noise
■	Uniform	Uniform noise
■	Tone	Sine function

Frequency. This field defines sine wave frequency in Hertz. You may enter a constant, or vary Frequency as a function of the SGI by assigning a SigGenRP variable.

Phase. This field specifies the sine wave onset phase or noise generator seed value at the start of each gating. You may use a constant for phase or vary Phase as a function of the SGI by assigning a SigGenRP variable.

Note: Default value for phase is 0, for Tone, this is zero phase, and for Gaussian and Uniform noises, this is continuous phase (phase changes continuously from trigger to trigger). You may wish to generate "frozen noise," noise that does not vary from trigger to trigger. To generate frozen noise, enter -1 in the Phase field.

Sweep Rate. You may wish to generate a sine wave sweep. To do so, enter the rate of frequency change (Hertz per second) in this field. You may vary Sweep Rate as a function of the SGI by assigning a SigGenRP variable.

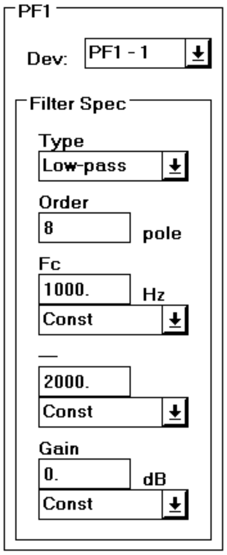
WG1 and WG2 can only provide limited sine wave sweeping capability, and Sweep Rate value must be between 1 and 10,000 Hertz per second

Mute While Updating

When this box is checked, output will be muted while parameters are being updated (between intervals).

PF1

SigGenRP supports programmable filtering through the use of TDT's PF1.



PF1

Dev: PF1 - 1

Filter Spec

Type: Low-pass

Order: 8 pole

Fc: 1000. Hz, Const

—: 2000., Const

Gain: 0. dB, Const

Dev.

You must specify the PF1 you wish to configure. This is accomplished by choosing the desired device from the Dev. list. You can select only one PF1 or None from this list.

Filter Spec

Type. This parameter allows you to specify the specific filter type. You may choose from:

- All-pass
- Low-pass
- High-pass
- Band-pass
- Band-reject
- Range-pass
- Range-reject

The parameters required to specify a filter vary depending on the chosen filter type. The required parameters for each filter type are presented below.

All-pass Filter

All-pass filtering allows the signal to pass through unfiltered. No parameter specification is required.

Low-pass and High-pass Filters

Order. This field specifies the number of poles necessary to achieve a user-desired roll-off rate. Roll-off rate equals 6 dB per octave per pole. For example, if you desire a roll-off rate of 60 dB per octave, you would specify 10 poles in the Order field (*60 dB per octave/ 6 dB per octave = 10 poles)*. This is a fixed number, and the maximum number of poles of the PF1 depends on frequency range of the filter. For example, when the corner frequency is above 1000Hz, the maximum number of poles is 10.

Fc. This field specifies the corner frequency (-3dB frequency) for low-pass and high-pass filters. You may vary Fc as a function of the SGI by assigning a SigGenRP variable. The accuracy (placement resolution) of the corner frequency (Fc) depends on the filter frequency range, see *PF1 Specification* for more information.

Gain. The value specified in the Gain field is used to adjust the signal level in the following manner: *Output voltage (volt) = Input voltage (volt)* × *Gain.* You may vary Gain as a function of the SGI by assigning a SigGenRP variable. This value must expressed in terms of dB.

Band-pass and Band-reject Filters

Order. See *Low-pass and High-pass Filters*.

Fc. When the filter type is band-pass or band-reject, this field specifies the center frequency of the band you wish to pass or reject. The accuracy (placement resolution) of the corner frequency depends on the filter frequency range. The you may use a constant value as Fc, or vary Fc as a function of the SGI by assigning a SigGenRP variable.

BW. In this field, you specify the bandwidth you wish to pass or reject. Bandwidth is defined in octaves. You may vary BW as a function of the SGI by assigning a SigGenRP variable.

Gain. See *Low-pass and High-pass Filters*.

Range-pass and Range-reject Filters

Order. See *Low-pass and High-pass Filters*.

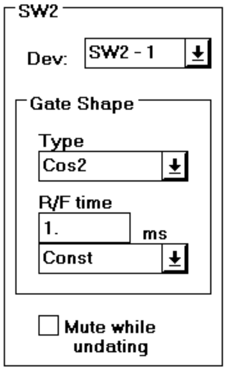
F1. This field specifies the -3dB lower corner frequency of the range you wish to pass or reject. You may enter a constant value, or vary F1 as a function of the SGI by assigning a SigGenRP variable.

F2. This field specifies the -3dB upper corner frequency of the range you wish to pass or reject. You may enter a constant value, or vary F2 as a function of the SGI by assigning a SigGenRP variable.

Gain. See *Low-pass and High-pass Filters*.

SW2

SigGenRP supports analog signal gating through the use of TDT's SW2.



SW2

Dev: SW2 - 1

Gate Shape

Type: Cos2

R/F time: 1. ms

Const

□ Mute while undating

Dev.

You must specify the SW2 you wish to configure. This is accomplished by choosing the desired device from the Dev. list. You can only select one SW2 or None.

Gate Shape

Type. This parameter allows you to specify the rise/fall shape of the gate. You may choose from:

- OnOff (no gate)
- Cos2
- Cos4
- Cos6
- Ramp
- Ramp2
- Ramp4
- Ramp6

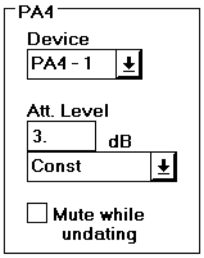
R/F Time. This field specifies the gate's rise and fall time in milliseconds. You may select a constant value, or vary R/F Time as a function of the SGI by assigning a SigGenRP variable.

Mute While Updating

When this box is checked, output will be muted while parameters are being updated.

PA4

SigGenRP supports programmable attenuation through the use of TDT’s PA4.



PA4

Device

PA4 - 1

Att. Level

3. dB

Const

□ Mute while undating

Device

You must specify the PA4 you wish to configure. This is accomplished by choosing the desired device from the Device list. If you choose the same PA4 as you do in the Signal Parameter dialog box, this selection will overwrite the other selection.

Att. Level

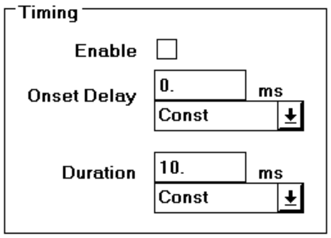
The attenuation level in dB is defined in the Att. Level field. You may use a constant value, or vary the attenuation level as a function of the SGI by assigning a SigGenRP variable.

Mute While Updating

When this box is checked, output will be muted while parameters are being updated.

Timing

You may control the overall timing characteristics of peripheral devices.



Timing

Enable □

Onset Delay 0. ms
Const ⭳

Duration 10. ms
Const ⭳

Enable

Checking the Enable box enables the timing parameters, Onset Delay and Duration.

Onset Delay

You may define an Onset Delay. You may vary Onset Delay as a function of the SGI by assigning a SigGenRP variable.

Duration

The Duration field specifies the signal duration. You may vary Duration as a function of the SGI by assigning a SigGen variable.

Appendix D Index Variable Example

To use five different frequencies in a tone pip experiment (index.sig in c:\tdt\siggenrp\examples) where each tone is calibrated to account for transducer variability across frequency you would need to use an index variable.

The tones for this example are listed below:

Frequency	**Transducer Sensitivity**	**Attenuation Setting**
1000 Hz	98.0 dBSPL/vrms	8.0
1400 Hz	96.5	6.5
2000 Hz	93.8	3.8
2700 Hz	92.0	2.0
3500 Hz	91.1	1.1

To get the tone pips to play out at a calibrated level the low frequency tones must be attenuated so that they match the level of the highest frequency tone. The last column shows some attenuation settings that could be used to normalize the transducer to have a uniform sensitivity of 90.0dBSPL/vrms.

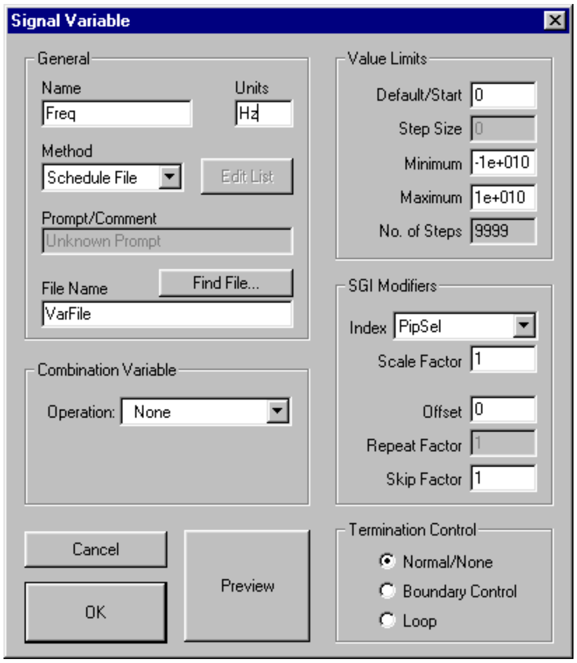
First make a variable file with the information for both frequency and the corresponding calibration attenuation:

```
{

Freq
1
1000
1400
2000
2700
3500
}
{

CalAtten
1
8.0
6.5
3.8
2.0
1.1
}
```

Next make a variable called *PipSel* that goes from 1 to 5, which will be used as the SGI modifier. For example, to do a frequency intensity series another variable called *Level* that steps through the desired levels is needed.

Next nest these variable using the Index modifier settings.

Now make two variables called *Freq* and *CalAtten* both using the variable file above as their source. Under SGI modifiers select the *PipSel* variable with a scalar of 1.0. These variables will then use the value of *PipSel* to pick which frequency and attenuation to use.



Signal Variable

General

Name: Freq

Units: Hz

Method: Schedule File

Prompt/Comment: Unknown Prompt

File Name: VarFile

Combination Variable

Operation: None

Value Limits

Default/Start: 0

Step Size: 0

Minimum: -1e+010

Maximum: 1e+010

No. of Steps: 9999

SGI Modifiers

Index: PipSel

Scale Factor: 1

Offset: 0

Repeat Factor: 1

Skip Factor: 1

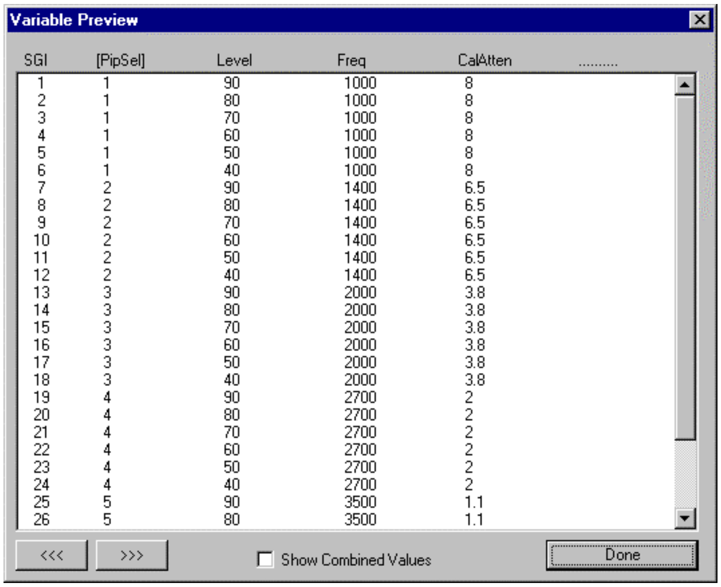
Termination Control

● Normal/None

○ Boundary Control

○ Loop

Now click preview variables and at the bottom on the screen click the combined values button.



Variable Preview

SGI	[PipSel]	Level	Freq	CalAtten	....
1	1	90	1000	8	
2	1	80	1000	8	
3	1	70	1000	8	
4	1	60	1000	8	
5	1	50	1000	8	
6	1	40	1000	8	
7	2	90	1400	6.5	
8	2	80	1400	6.5	
9	2	70	1400	6.5	
10	2	60	1400	6.5	
11	2	50	1400	6.5	
12	2	40	1400	6.5	
13	3	90	2000	3.8	
14	3	80	2000	3.8	
15	3	70	2000	3.8	
16	3	60	2000	3.8	
17	3	50	2000	3.8	
18	3	40	2000	3.8	
19	4	90	2700	2	
20	4	80	2700	2	
21	4	70	2700	2	
22	4	60	2700	2	
23	4	50	2700	2	
24	4	40	2700	2	
25	5	90	3500	1.1	
26	5	80	3500	1.1	

☐ Show Combined Values