SCOPE /SDK

Version 4.0

Quick Start

CreamWare Audio GmbH
Wilhelm-Ostwald-Strasse 0/K2
53721 Siegburg
Germany

Tel.: (+49) 2241-5958-0

Fax: (+49) 2241-5958-57

Hotline: (+49) 2241-5958-12

> Main Table of Contents

The CreamWare
SCOPE /SDK *Quick Start*

Welcome to the SCOPE /SDK Quick Start. This guide will familiarize you with the basic steps for using your new audio development suite. It will give you a brief overview of the different functionalities of the SCOPE environment and it will set you up for your first break up.

We wish you a good start into the possibilities of your new DSP development environment.

Although you might feel ready to use the development suite by having just worked through the Quick Start we want to point out that it is highly recommended to read the User Manual as well as the different tutorials thoroughly. To take full advantage of the features and the power of the SCOPE /SDK we want to encourage you read the User Manual properly.

! *Please note:*
For your convenience we used small capitels so that you can easily recognize modules in the text.

This Quick Start is divided into two parts. The first one gives you a short introduction to the environment of SCOPE /SDK.
In the second part you will learn how to perform basic tasks in SCOPE /SDK. Therefore we will build a small circuit- an input detector. We will learn about building circuits, using interfaces. organizing our work, building simple interfaces and editing them.
We will then write a small script so we will learn basics on how to enlarge the powers of SCOPE /SDK on our own.

When done with the Quick Start you are prepared to untertake your first self-employed experimentations with the SCOPE /SDK.

Contents

The CreamWare SCOPE /SDK Quick Start 2
.... 2
Part I: Setting up SCOPE 4
Starting the SCOPE /SDK application and adjusting the SCOPE display settings 4
The SCOPE desktop 5
Adjusting the I/O settings 8
.... 8
Output a signal – Building a simple circuit.. 8
Part II: Working with SCOPE.. 9
.... 9
Using the interfaces of basic modules 13
Extending the circuit –Building an input indicator 14
Organizing a circuit 19
Adjusting Surface Controls 21
Layout - Designing our own surface 23
.... 25
Index 27

Part I: Setting up SCOPE

Starting the SCOPE /SDK application and adjusting the SCOPE display settings

When you installed the SCOPE /SDK software package successfully you will find an alias on your computer desktop. If you have not installed the components yet or you came across any difficulties concerning the installation, please consult the Installation Guide for further information about the installation procedure and trouble shooting.

Double-click the SCOPE icon to start the SCOPE /SDK.

You can also start the application from the Programs menu of your Windows operating system. (click the 'Start' button in the device bar, open the 'Programs' drawer , go to the CreamWare and menu start the SCOPE /SDK application).

The SCOPE desktop pops up.
Principally the SCOPE desktop should look familiar to you. It has a menu bar at the top with a Minimize and a Close button – just like any Windows program.

We want to draw your attention to the fact that the SCOPE /SDK is not displayed in window that can be resized, like most other applications. The SCOPE /SDK is a platform on its own – like an operating system running on top of the Windows OS. The SCOPE fusion platform uses its own routines (like copy, paste, drag and drop, etc.) and is therefore platform independent.

The SCOPE /SDK may not display adequately.
This is because your monitor resolution differs from the default resolution of SCOPE.

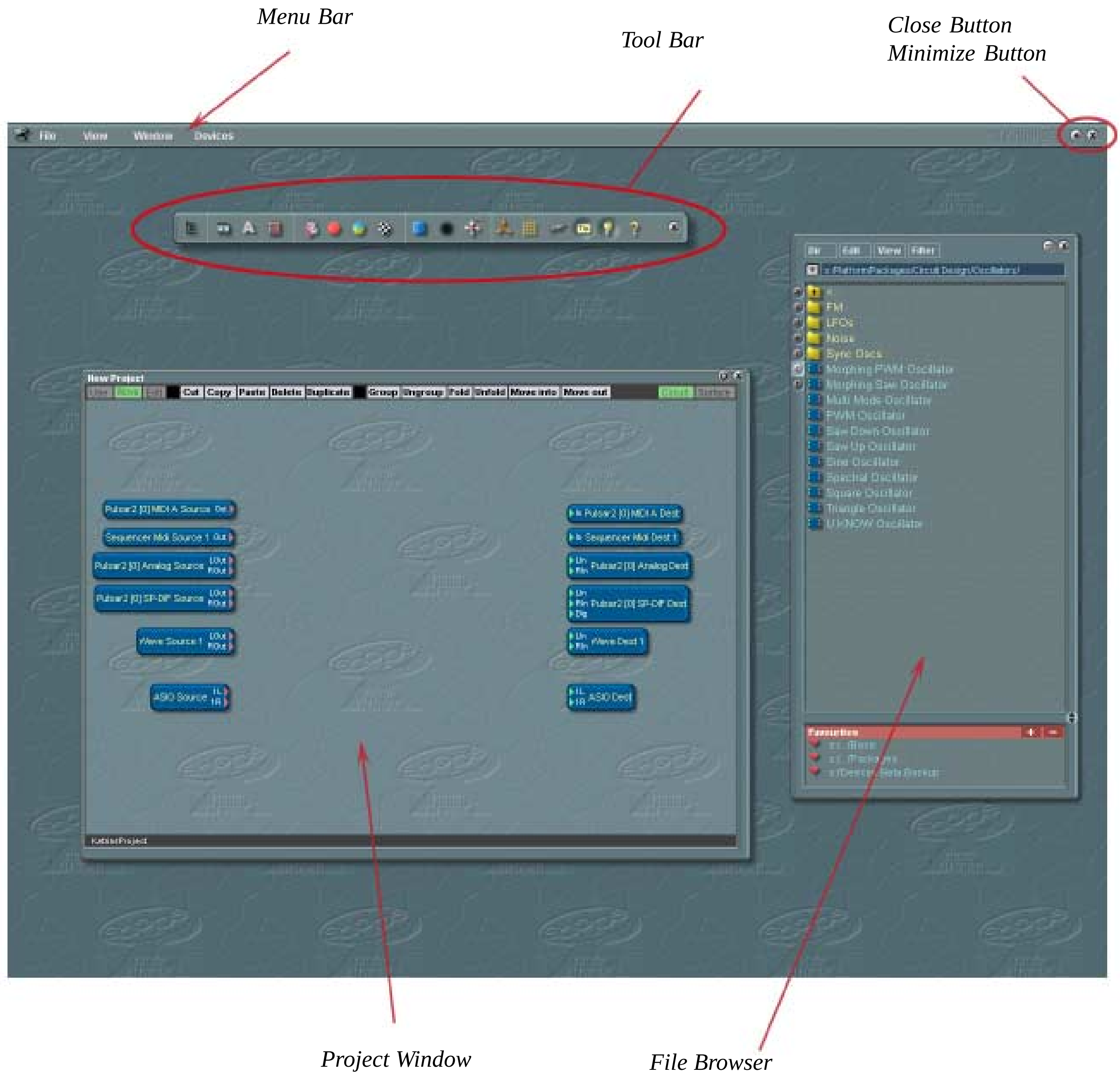
To adjust the display settings open the *SCOPE Settings* dialog box from the *menu bar* **(View -> Settings)** and double-click the mini view of the SCOPE desktop on the right half of the dialog box. Then close it. The SCOPE desktop will now fit to the screen.

This dialog box also gives you the option to configure SCOPE for dual screen systems.

To learn more about the SCOPE display settings please read the chapter 'lurking' in the User Manual.

The SCOPE desktop

On the desktop you see the *Project Window,* the *File Browser* and the *Tool Bar.*



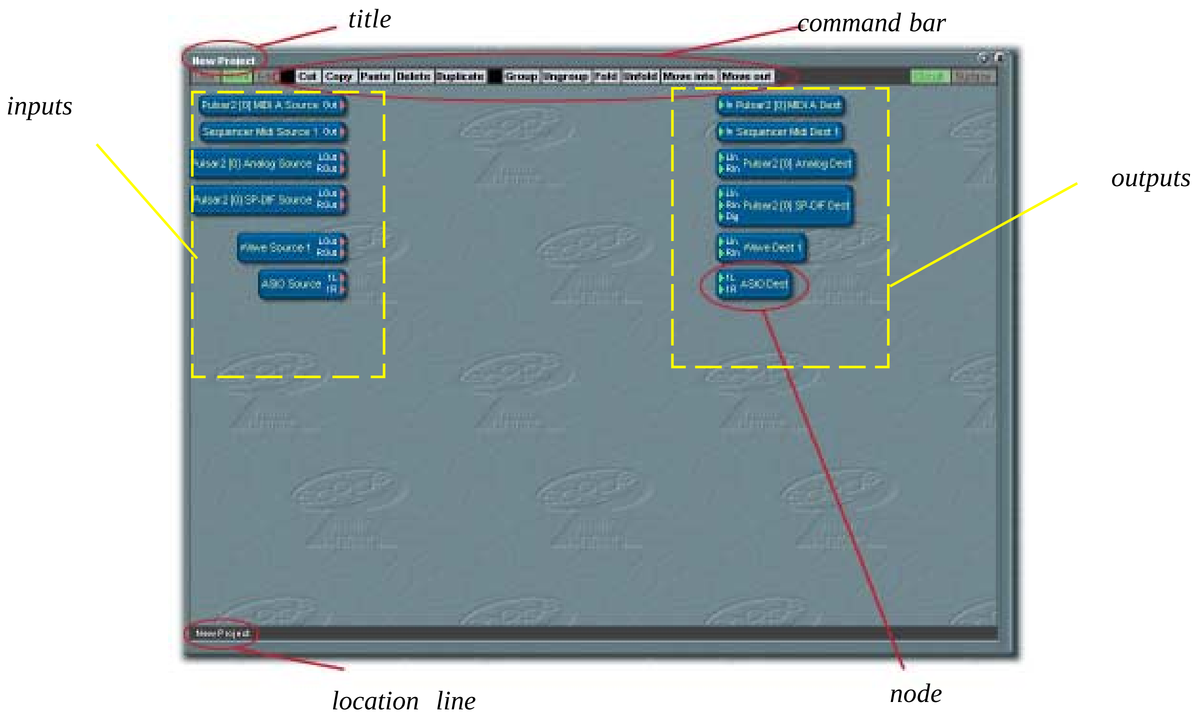
The *Tool Bar* gives you access to a lot of tools that will help you to customize the design of your own surface. It is comparable to the 'Palette' in some graphic programs. We will have a closer look on them later on.

The *Project Window* is the actual workspace of the SCOPE /SDK. In this window you layout the design for the processing circuits as well as the interfaces for your devices.

The workflow throughout SCOPE is a project oriented approach. Therefore the file type is a project file. It contains information about the used I/Os, the used modules and their parameter settings.
As you can address the I/Os (I/Os are physical and cannot be virtually duplicated) only once, you have to work on one project at a time. Otherwise you would get an I/O conflict.

! ***Nodes*** *are circuit representations of entities. Entities can serve different needs – for example they can generate or process audio signals.*

Their names indicate which port they refer to (like Scope ADAT A Source is the representation of the first physical ADAT input on your SCOPE board). On the left side there are nodes called 'Sources' that stream data into the SCOPE environment. The nodes on the right side stream data out of the environment to other components. They are labeled 'Dest' for destination.



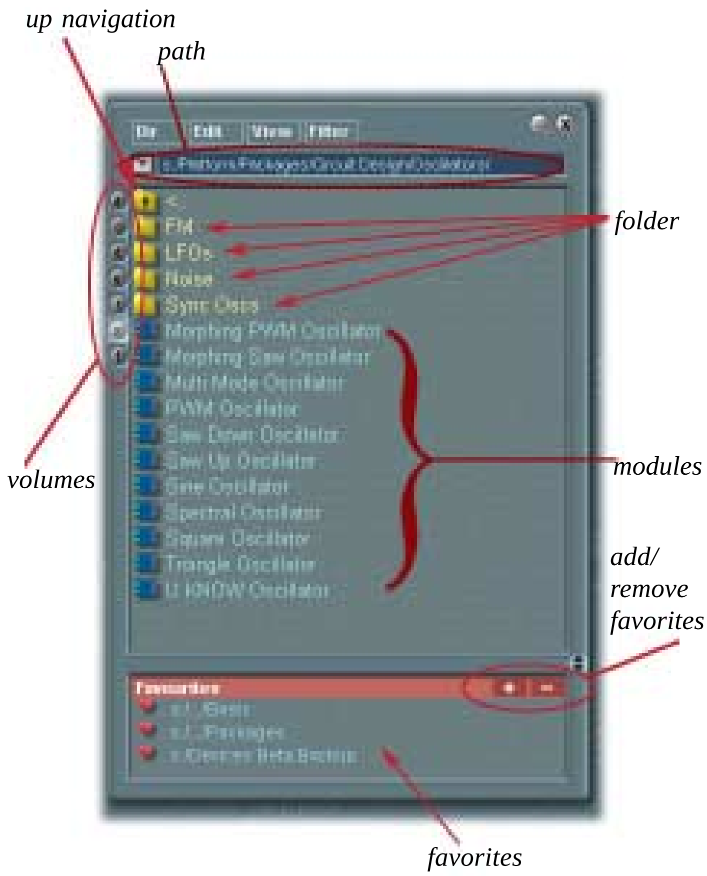
The name of the project you currently work on is indicated by the title of the *Project Window.*

The *location line* at the bottom indicates which hierarchy of the project is displayed. Inside the *Project Window* you can see some blue boxes – these are the I/O nodes.

If you look closely you can distinguish two different kinds of I/Os. On the one hand there are software I/Os. These handle the communication with programs and routines which are part of your operation system. (Examples for the software I/Os are Wave Source/Dest 1 and Sequencer Midi Source/Dest 1.) On the other hand there are hardware I/O nodes which are the counterparts of the physical connectors of your CreamWare DSP board.

! *Please note that the actually displayed I/O nodes depend on the installed I/O-Plate and your DSP board. To learn more about the different I/Os and DSP boards consult the Hardware Description Guide.*

The third element on your SCOPE desktop is the *File Browser.*



As the name suggests it gives you access to the volumes of your computer. From the *File Browser* you can drag and drop modules and devices into the *Project Window*.

It provides a fast and easy way to load files into the SCOPE /SDK application.

! *A **module** is made up of atoms and/or modules. Each module can consist of a number of other modules. A module can but does not necessarily need to have a surface.*

! *A **device** is a complete working design including a graphical surface, controls, circuits etc. Examples of devices are the various synths, mixers, and effects modules.*

! *An **atom** is the smallest entity holding executable code. There are four different types of atoms DSP-Atoms, PeP-Atoms, VxD-Atoms, DLL-Atoms.*

At the bottom of the *File Browser* window there is a section to save, manage and access our favorites.

It is easy to create bookmarks for the different locations of your files on the multiple volumes of your computer.

Select the directory you want to bookmark in the *File Browser* and just click on the plus sign '+' at the top right of this section. If you want to remove one of your favourites you can click on the minus sign '-' and the bookmark link disappears from your *Favorites* list.

Adjusting the I/O settings

We now want to assemble a little circuit and feed some audio signals to the outputs. In case you have only digital connectors (e.g. the ADAT plate) or you want to use digital connectors for some reasons, you have to assure that the I/O settings are correct.

The SCOPE system should be synchronized to your outboard equipment. Open the *Sample Rate Settings* dialog box (**View -> Sample Rate**) and select the appropriate Master and/or sample rate. Depending on the used equipment, SCOPE can either serve as Clock-Master or as Clock-Slave.

Adjust the settings and close the dialog box.
You do not need to restart SCOPE.
Now connect one of SCOPE’s outputs to your monitoring.

! *It is advisable to turn down the volume while setting up the connection. Make sure that everything is wired up correctly.*

Part II: Working with SCOPE

Output a signal – Building a simple circuit

Now we want to create our own circuit by putting together a couple of different modules.
Normally a basic circuit consists of at least two units: one generating a signal and another doing any kind of processing.
The processed signal is then passed to the audio outputs.

The *Project Window* is the place where you design the circuit. You build up a circuit by generating a network of basic modules.

SCOPE /SDK offers a huge library of ready to use basic modules which can be found inside your installation folder.

From the *File Browser* you can drag and drop these modules into the Project Window.

The term ***module*** *is widely used throughout the SCOPE environment. Speaking of a module can mean referring to a basic module as well as referring to a complex module that is build up of thousands of basic modules. You might specify a module as an entity that combines one or more algorithms in one context. Often a module can be re-used in a greater context .*

In the *File Browser* navigate to the SCOPE installation folder.

If you previously closed the *File Browser* or the *Project Window* you can open them by choosing the corresponding items in the *Window* menu of the *menu bar* (**Window -> File Browser** , **Window -> Project Window 1**).

You can also press function key F9 to bring up the *Project Window* and F10 to open the *File Browser*.

Inside your installation directory there is the ‘SCOPE library’.
Open the **Circuit Design** folder in the library.

The **Circuit Design** folder contains a huge variety of basic modules which make it easy to build up more sophisticated entities.

Go to the **Signal Adders/Mix** folder and drag Mix 1 from the **Mix** folder into the *Project Window*.
This is a controllable signal attenuator that will allow you to adjust the level of the outgoing signal. Therefore you might want to place it near the outputs which are connected to your monitors.

! *Note that the newly added node already has a simple interface that allows you to control it immediately.*

There are multiple *Pads* on the modules circuit representation. These show up as colored triangles.

! ***Pads** are representations of variables of the module which can be accessed from other modules or from control elements (like surface control elements: e.g. sliders, potentiometers, etc.). This means they can be connected to Pads on other nodes.*

The green *Pads* are the inputs of the module. The red ones provide processed or generated data for other modules – they are the outputs of the node. When you move the mouse pointer over one of these *Pads* the pointer changes to a connector and a small help tag displays the full name of the *Pad*.

The 'Gain' *Pad* at the top side of the Mix 1 node already has some sort of a connector.

This so called antenna indicates that the input is connected to another module.

In our case the antenna links the controls of the surface with the Pad on the Mix 1 node.

As a next step we want to add a module that generates a sound. Navigate back

up two levels inside the *File Browser* by clicking on the *up navigation* so that you are back on the level with the **Circuit Design** folder. Enter the **Oscillators** and drag in the Spectral Oscillator.
This module has a simple interface to control its parameters right, too.

On every surface, next to the label, you can find a *Close Group* which consists of the *'Keep on Top'* button and the *'Close'* button.

With the '*Keep on Top*' button you can determine whether the surface should

stay on top or not. As long as the button has the red bar displayed it will stay on top, even if another window behind it is activated.
Be aware that otherwise it may disappear behind ohter surfaces and windows. You can bring it back above the *Project Window* by selecting it.
With the '*Close*' button you can close this surface.

We now want to pass the signal of the Spectral Oscillator via the Mix 1 to our monitoring.

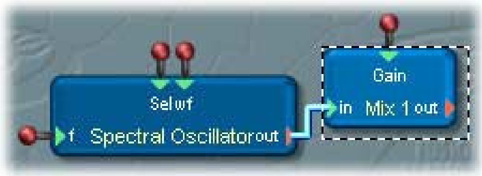
> ! *Before you stream an audio signal out of SCOPE onto your monitoring you might want to approve that the volume is turned down. In this case it is advisable to check if the gain of Mix 1 is still at its default value of '–24 dB'.*

To be able to hear the signal of the oscillator we have to chain up these two modules, the Oscillator and the Mix, with the output I/O that is connected to your monitoring system. This is done the in following way:

- Select the *Pad* at which you want to start a connection. Here it is the output *Pad* 'out' of the Spectral Oscillator.
  As mentioned before the mouse pointer will change to a connector as soon as you move it over a Pad or the Pad's name. In most cases you can establish connections from both sides.

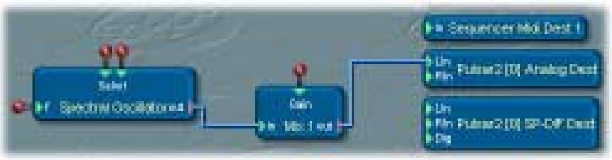
- Release the mouse button, drag the mouse over to the destination node and position the mouse over the *Pad* you want to connect to.
  In our case the destination *Pad* is the input *Pad* 'in' of Mix1.
  While the mouse is not over a Pad the mouse pointer is a red connector to indicate that you are currently dragging a virtual wire to another Pad. As soon as the pointer is over a Pad the connector will turn black again.

- Over the *Pad* or its name press the mouse button to establish the connection. A virtual wire will illustrate the connection.
  SCOPE will automatically prevent you from establishing any invalid connections. You can see this if you move the mouse over the output Pad of Mix 1 instead of over the input Pad before making the connection.



- To release a connection select the virtual wire with the mouse and press the delete key 'Del' on your computer keyboard.
  The wire will become highlighted if it is selected. Alternatively you can either click on the Delete button in the command bar or re-select the two connected Pads. When moving the mouse over the second Pad the pointer changes to a scissor which enables you to cut the wire symbolically.

If you have followed these steps the two *Pads* should now be connected via a blue line. Repeat this procedure to connect the output of Mix 1 to your appropriate output. As soon as you released the mouse button from the second Pad you should hear a sound. Our circuit should somehow look like that:



Depending on how you have set up your signal it is only present on one of the n-channels of your monitoring. This is the case if you are set up to output a stereo signal on a stereo listening.

Assume you want the signal to be present on two adjacent channels. You could now connect the output of the Mix 1 to the second output channel.

As this project is very handy and the nodes are placed next to each other doing this is no problem at all. First click on the one, then on the other *Pad* and another virtual wire wil appear.
Nevertheless when working with huge projects it is more comfortable connecting the first *Pad* on the I/O node with the second one.The signal from Mix 1 is now present on the first two outputs.

Save this project by using the 'Save as' dialog box from the File menu.

It is a good idea to save your files from time to time. You may also want to save a new copy of the project with each session successively just in case you want to go back in work history.
You could use an index in the name of the project and augment it for every session.

Using the interfaces of basic modules

A basic module has an interface if it has adjustable variables. Each one has an easy-to-use surface control element (like a slider, a button, a control text field etc.) on the module's surface.

The parameters might have a high resolution. Generally this gives you the possibility to really make subtle changes but it might seem hard to select a specific value within the overall range.

For example, when changing the frequency of the SPECTRAL OSCILLATOR by using the potentiometer you might get coarse changes making it hard to fine tune the pitch.
When you now click on the potentiometer that controls the frequency and move the mouse towards the edge of the screen the changes become finer.

To make the adjustment of the values as comfortable as possible you can also type the exact values into the text fields. (Just click on the text field, type the value and press the 'Enter' key on your keyboard).

The waveform text field of the OSCILLATOR is actually a text-based fader. So you can change it by clicking into the field und dragging the mouse up or down (while keeping the mouse button pressed).

You may now want to change the different parameters to see how the control elements work und how they impact the signal. You can make fast and coarse changes or you can fine adjust the parameters with the outlined techniques. Double-clicking on the potentiometer will set them back to their default value.

If you have closed an interface and want to re-open it, locate the corresponding circuit representation of the module. Right-click it and a *context menu* appears. From the *context menu* select **Surfaces -> Open Panel**. The interface will pop up.

Extending the circuit – Building an input indicator

If you listened to the signal while you were doing the modifications in the last chapter you heard how the output of the SPECTRAL OSCILLATOR changed.
Perhaps you are not quite sure what exactly is happening when you modify the 'Modulation' parameter and you simply like to have a visual feedback.

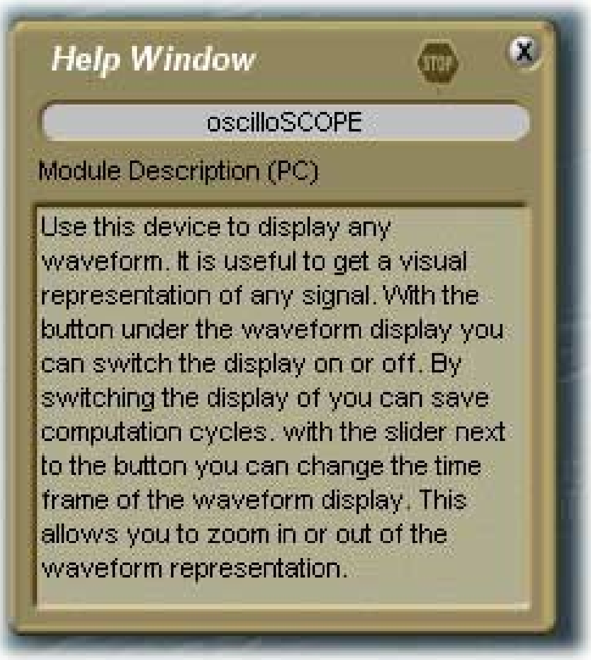
For the above mentioned needs we go to the *File Browser*. On the same level as the **Circuit Design** folder we find the **Tools** folder. It provides a simple oscilloscope that will display the waveform. Drag it into the *Project Window* und connect its input to the SPECTRAL OSCILLATOR's output.

The OSCILLOSCOPE offers you the possibility to adjust the level of the incoming signal and it has a VU meter to indicate the level. With the horizontal fader beyond the waveform screen you can scale the time-frame for the visualization.

If you make any changes to the OSCILLATOR`s parameters you will get a audible and visual feedback.

Now we want to build a module that outputs a sound when it detects a signal on its input. At which level of the input signal the module should output a sound will be adjustable.

By creating this module we will come across a lot of basic modules and even more *Pads*. SCOPE offers a *Help Window* which displays basic information about the different modules and their Pads.



To open the *Help Window* click on the question mark in the *Tool Bar* or press F1. It also provides information about the SPECTRAL OSCILLATOR, the MIX 1 and the OSCILLOSCOPE. So if you open the *Help Window* you can just move the mouse over the different modules and get some helpful information.

Perhaps it is a good idea to analyze the components and structure of this module which we want to build, more precisely: The module has one input and one output. We have a signal A that will be heard as long as a signal B at the input is present and its level is higher than a specified threshold. So we need a component that detects if the level of signal B is higher then specified. In that case it has to trigger the output of signal A. That is pretty much straight forward.

We already have the signal generator for signal A – the SPECTRAL OSCILLATOR. That is connected to an attenuator to control the volume of the output signal.

First of all we need a component that detects if there is a signal B with a specific level. It also should send out a recognizable message if this is true. It will trigger when the amplitude of the incoming signal is higher then the threshold.

Drag in the AUDIO TRIGGER module (**Circuit Design/Gates/AUDIO TRIGGER**). The AUDIO TRIGGER compares two signals with each other or a signal to a given value (threshold).
We will use the second option.

The problem of this node is, that it takes the actual signal and not the level of the signal. So it will retrigger all the time.

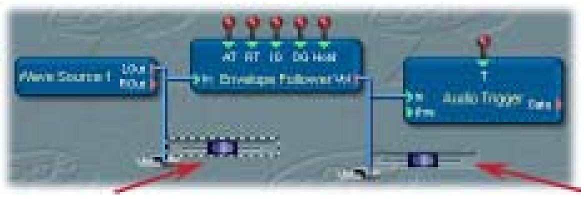
This is not exactly what we want. First we need to get the level of signal B. Then we can input the resulting values to the AUDIO TRIGGER. We can achieve this by taking the ENVELOPE FOLLOWER (**Circuit Design/Effects/Others/ENVELOPE FOLLOWER**) which derives the level of the incoming signal and outputs it as control values.

We connect the 'Vol' output of the ENVELOPE FOLLOWER with the 'in' *Pad* of the AUDIO TRIGGER. Leave the 'Source' parameter of the AUDIO TRIGGER set to 'Value' (the text-based fader on the surface).

To illustrate why we have inserted the ENVELOPE FOLLOWER drag two Faders (**Circuit Design/Circuit Controls/FADER 1 HORIZONTAL**) into the *Project Window*. Position the first fader (FADER 1) behind the ENVELOPE FOLLOWER and the second one (FADER 2) below the FOLLOWER.

Connect the output *Pad* 'Vol' of the ENVELOPE FOLLOWER with the 'Val' *Pad* of FADER 1 and connect the output of a sound source with FADER 2, e.g. the output of WAVE SOURCE 1. Now you will see how the ENVELOPE FOLLOWER works.

If a signal is coming in, the two faders start to move but you will see a difference between them: FADER 2 moves more frequently than FADER 1.



Fader 2 *Fader 1*

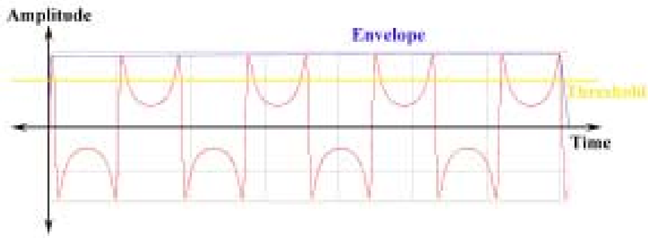
FADER 1 will follow the envelope of the signal whereas FADER 2 follows the waveform itself.

We can also visualize this graphically:

The red curve is the incoming signal. It has positive and negative values and it crosses the threshold of the AUDIO

TRIGGER several times.Therefore the AUDIO TRIGGER retriggers with each transition.

The ENVELOPE FOLLOWER only generates positve values as indicated by the blue



envelope line. It crosses the threshold only two times: at the beginning and at the end of the incoming signal.

Now we have two different sub-networks:

One that generates signal A and the other one deals with signal B. In the consecutive step we want to bring the two circuits together.

Focusing on signal A, we only want to output it if signal B exceeds a specific level. At the moment the oscillator always generates and outputs the signal. So we need something like a gate that opens only if the our condition holds.

Instead of a real gate we could use an envelope in conjunction with a volume attenuator. The envelope is triggered by the AUDIO TRIGGER which has a Gate output *Pad*.

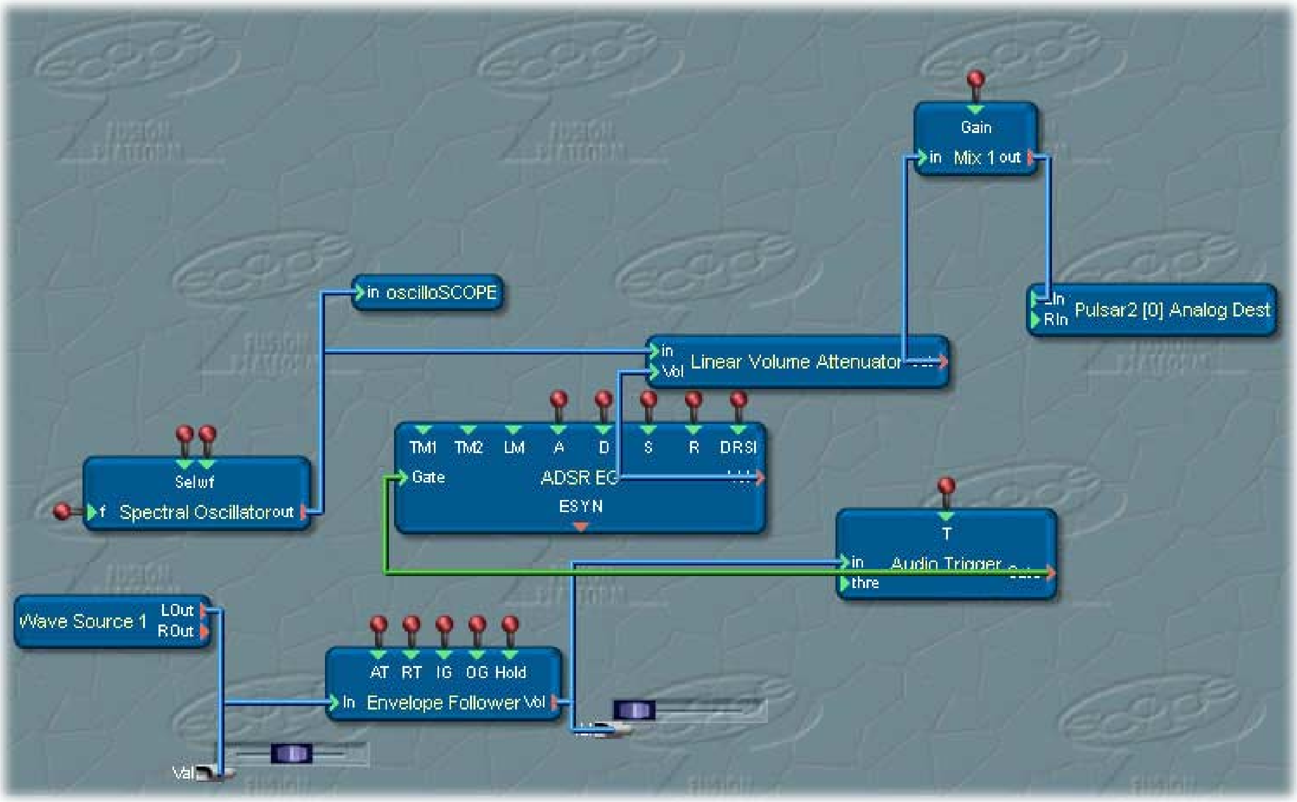
Place a LINEAR VOLUME ATTENUATOR (**Circuit Design/Envelopes/Linear VOLUME ATTENUATOR**)between the SPECTRAL OSCILLATOR and the MIX 1 attenuator.

Delete the current connection and connect the 'out' *Pad* of the OSCILLATOR with the 'in' *Pad* of the LINEAR VOLUME ATTENUATOR (LVA). Then the 'out' of latter with the 'in' of the MIX 1.

From the same folder place an ADSR ENVELOPE GENERATOR (ADSR EG) between the AUDIO TRIGGER and the LVA. Wire up the 'Gate' *Pads* (AUDIO TRIGGER – ADSR EG) and the 'Vol' *Pads* (ADSR EG – LVA).

If you connected the basic modules as lined out it should somehow look like that:

Now delete the connections between the WAVE SOURCE, the FADERS and the ENVELOPE FOLLOWER. (As mentioned before: Just select the wire and it will become highlighted. Click on the *Delete* button in the *command bar.*)



To generate the signal B that feeds our network we will use the Test Generator (**Circuit Design/Basics/Test Generator**) which is helpful every time you need some test signal. Connect the output of the Test Generator to the input of the Envelope Follower. On the Test Generators interface switch on the sine waveform. Adjust the parameters of the different components.

Here are some example values for the most important Parameters:

Test Generator
Sine Generator: on
Gain: -16 dB

Envelope Follower
Attack: 0.1 ms
Release: 900 ms
Input/Output Gain: 0 dB
Follow

Audio Trigger
Threshold: 8
Source: Value
Sensitivity: 0

Mix 1
Gain: 0 dB

The most interesting values are the ‘Gain’ value of the Test Generator and the ‘Threshold’ value of the Audio Trigger. Changes to them will determine if signal A can be heard or not.

With the shape of the envelope (ADSR EG) you can adjust the time response of the outgoing signal.

You may also want to feed signal A into the OSCILLOSCOPE. The VU meter and the waveform screen provide you with a visual feedback. You may now want to input other signals into the network – use the WAVE SOURCE node or an external input.

Furthermore we can delete the FADERS and the WAVE SOURCE because we do not need them anymore.

Organizing a circuit

Relying on a network to build modules has some great advantages. You can easily break connections, try other routings, start a new chain anywhere or insert new modules wherever you want.

The disadvantage is that it can get confusing which makes it difficult to handle large networks.
Therefore you are able to fold modules and networks. This helps to structure a complex module by grouping nodes into sub-networks and finally folding this structure so that the module is represented as a single node.

Select the nodes that belong to our module (**Envelope Follower, Audio Trigger, ADSR EG, LVA, Spectral Oscillator** and the **Mix 1**) by drawing a rectangle around them and press the *Fold* button in the *command bar* of the *Project Window* (alternatively you could use the shortcut 'Ctrl-F').
The nodes are folded into a single one. All the connections are still intact.

Now your circuit is easy to survey:

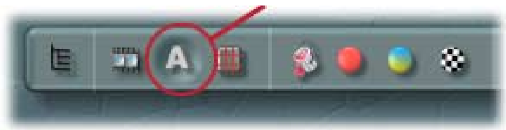
! *You can select multiple nodes by drawing a rectangle around them or by shift clicking one after the other. If you left one out select the newly created node and press the unfold button in the command bar. Reselect the specified ones and fold them again.*

The name of the node does not really meet our expectations. Neither describes it the functionality of the

module nor is it readable.

Open the *Text* tool from the *Tool Bar* (press function key F12 to show the *Tool Bar*) by clicking the '**A**'

Text tool icon



pictogram.

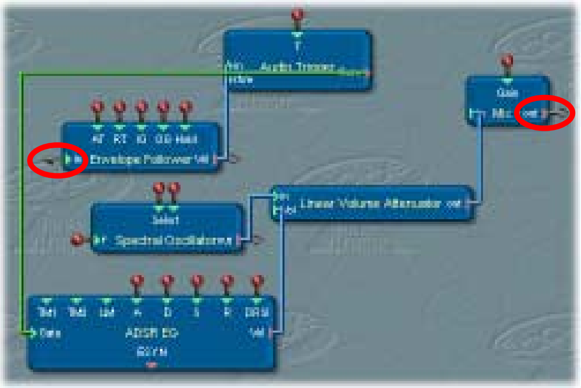
Select the node you want to rename. The name appears in the text tool dialog. Type in a name like 'Input Detector'.

You can easily edit your circuit by double-clicking the wrapping node. This way you step one layer deeper into the hierarchy of a module.
The number of layers a module can have is not limited. To return to the upper layer double-click in the background area of the *Project Window*.

The navigation through the layers of modules is:

- double-click on a node to step one layer deeper
  double-click on the background of the Project Window to return to the upper level

Enter the folding node so that you can see the circuit. You might have recognized that there is a sort of arrow on the input of the Envelope Follower and the output of the Mix 1. These are the *Pads* that are connected to other modules in the network; and as we have mentioned before these connections are still intact.



An arrow indicates that a *Pad* has been exported. So the *Pads* on the wrapping node are equivalent to the *Pads* on the basic modules.

! *When folding, existing connections stay intact and the Pad is automatically exported and displays on the wrapping node.*

In our circuit the automatic exportation causes the MIX 1 output *Pad* to appear twice on the folding node. This happens because we have two modules connected to it – the I/O node and the OSCILLOSCOPE. This does not really cause a problem but it could be confusing. To change this we select the 'out' *Pad* of the MIX 1 and from the right mouse button *context menu* we choose 'Delete exported Pad'.
The arrow will disappear as will its two representations on the outer node. Of course this will also break the connections.

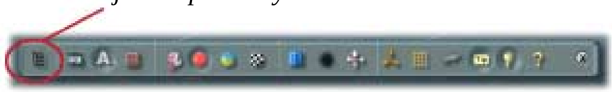
Export the *Pad* again by using the equivalent command from the *context menu* und reconnect the module to the I/O node and the OSCILLOSCOPE.

At the bottom of the *Project Window* in the *location line* SCOPE shows the path of the currently displayed layer.

A more convenient overview of the project is provided in the *Project Explorer*. Like the Windows Explorer it represents the structure of a project as a hierarchical tree. The way you use it is equivalent to the Windows Explorer, too.

To open the *Project Explorer* click on the symbol on the *Tool Bar* or use the Window menu in the *menu bar* (**Window** -> **ProjectExplorer**). You can as well press F11 to open the *Project Explorer*.

Project Explorer symbol



In the *Project Explorer* modules are displayed as a blue box. A folding node additionally has a '+' sign inside the blue box. By clicking on the box the node will display its content.

When you select a node in the *Project Window* it will become highlighted in the *Project Explorer*. The tree will automatically expose the path to this node by opening the wrapping ones.

Although we have renamed our module to 'Input Detector' the *Project Explorer* still shows the old name. To change that, select the old name in the *Project Explorer* and press function key F2. The name can now be changed to 'Input Detector' .

! *F2 is the universal edit command throughout SCOPE.*

It is a good idea to save the project now.

Adjusting Surface Controls

Now we have a single INPUT DETECTOR module. We must state that it has a couple of interfaces making it hard to control each and every parameter. But after having adjusted them once we actually need only one surface – that of the AUDIO TRIGGER. So we can close the other ones.

When using the module it soon becomes clear that the parameterization of the threshold controls (of both, the text field and the potentiometer) do not fit our needs. It would be nicer if the parameter could be controlled in dB.

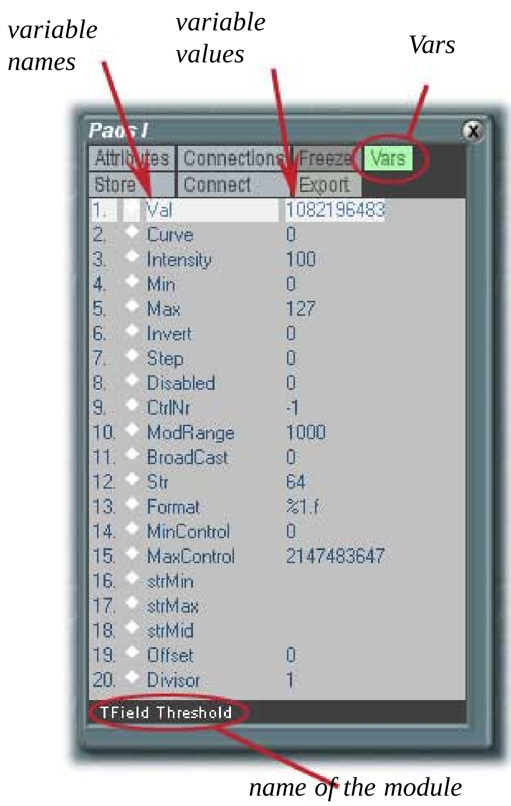
Make sure the *Project Explorer* is open. At the top of it there are different window access buttons. Click the *Pad I* button. A new window will pop up. This is the *Pad List* window.

It lets you edit the *Pads* and variables of a control element or a module. Now click in the text field of the AUDIO TRIGGER.

! *SCOPE distinguishes between* ***Pads*** *and* ***Vars*** *(variables). Vars pass parameters to a module (a surface control element is a specific form of a module).*

! ***Pads*** *are references of* ***Vars****. They can be accessed from other modules. Pads have an external representation that allows them to be connected. So it is possible to have multiple Pads for each variable. If you need a variable to be accessible from another module you have to export a Pad for it.*

If you have selected the text field of the 'Threshold' control you will see one *Pad*. At this point we do not need to worry about this *Pad*. We will straight go to the *Vars* of the text field by activating the *Vars* view of the *Pad List*. This can be done by clicking on the *Vars* button in the upper right corner of the *Pad List* window.You will see a list of variables. These are the variables that determine the behavior of the text field. We will now make the text field behave like a level control.



First of all we will change the format of the scale to dB. In the list you find a format variable '**13. Format %1.f**'.
Now we have to change the value of the format variable from '**%1.f**' to '**%1.1f dB**'. To edit this value select it and press the function key F2 and type in the new value.To confirm it press 'Enter'.
You will see, that the text field displays the first digit after the point as well as the format dB.

Having this done, the range of the text field goes from 0.0 dB to 127.0 dB. This is very unlikely for a level scale. It should rather go from '**– inf**' to '**0 dB**'. If we have a look at the Max and Min variables (numbers 5 and 4) you can guess why this is the case.

For the level scale we are approaching the values should be **–1866** for **Min** and **0** for **Max**. So we select the values of those variables, press F2 and type in the new values.

To make things even better we enter a **Divisor** of **10** (this is the last variable, number 20). As we have the option to set a string for the **Minimum** value (nr. 16) we type in '**- inf**'.

Having completed these steps the text field is actually a lot more convenient than before. The value is still changing linearly.

Decibel is a logarithmic relation. So we need the text field to output an exponential curve (indeed this means that the text field takes a logarithmic curve as input and decibel is a logarithmic relation).

In order to achieve this, we change the value of the variable **Curve** to **2** and the value of the variable **Intensity** to **2147483647**.(This is the highest value that can occur in SCOPE. It is 2^{31}-1 which is equivalent to 31 bit.)

So the variables of the text field you actually changed are:

13. Format: %1.1f dB
4. Min: -1866
5. Max: 0
20. Divisor: 10
16. strMin: -inf
2. Curve: 2
3. Intensitiy: 2147483643

Now click on the potentiometer and you will see the variables in the *Pads I* window.
The potentiometer alters the level analog to the amplitude of the signal. When it is in the middle position we have a value of '–6dB'. Although we want to have a higher resolution as the level approaches the value of 0 dB this resolution is much to high.
A value of –18 dB for the middle position would be appropriate for the potentiometer. To achieve that, we have to assign a logarithmic curve to the control element.

Change the values of the variable **Curve** and variable **Intensity** of the control to **1** and **48**. This will give us fine control over the threshold.

After these changes our Input Detector is easier to use. Once more we should write our progress to disk. Save the project.

Layout - Designing our own surface

After we successfully adjusted the control elements, we can concentrate on the design of our own surface by editing the surface name and changing the colors.

As mentioned before the Audio Trigger surface is the surface for our new module. The problem with this interface is, that we can not access it from the *context menu* of the node because this way only surfaces on the first layer show up in the *menu*. But the interface of the Audio Trigger is on the second layer. In order to change this take the *Project Explorer* and navigate to the Audio Trigger panel. (The fastest way to get there is to select the panel.)

Select the SurfaceInterface and drag it onto the Input Detector node.
Do the same with the Panel node. Now our module has its own surface which can be opened and closed through the *context menu*.

The label of this surface does not correspond to the name of our module. To change it we have to close the surface and step one layer deeper. There we will find the Panel node.

Switch to *Surface* view to be able to edit the surface. Either press the button in the right upper corner of the *Project Window* or press F8.

The Project Window provides two different views - Circuit view and Surface view. The current view is indicated by the buttons in the upper right corner of the Project Window. You can toggle between the views by pressing the buttons or using the shortcuts F7 (Circuit view) and F8 (Surface view).

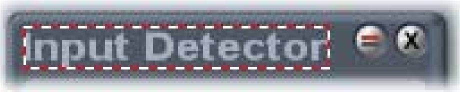
! *In **Circuit view** the circuit is displayed. This is where you add nodes and make connections from one module to another to route Audio, MIDI and control streams.*

! *In **Surface view** you design the user interface for the devices. There is a decoupling of circuits and surfaces so that you can really concentrate on designing the interface without having to handle all the circuit nodes in the same window, too.*

The surface will show up in the *Project Window.*
Now we have to switch to the *Edit* mode.

*For both views there are three operating modes - **Edit** ('Ctrl-E'), **Move** ('Ctrl-M') and **Use** ('Ctrl-U') mode. The three buttons in the upper left corner of the Project Window indicate which mode is currently active. You can also switch between the modes by using these buttons.*

*In **Use** mode we can alter surface controls. We can also establish connections between Pads like in **Move** mode. There we can additionally layout our circuit and navigate through the layers. In **Edit** mode we can edit the representation of circuit and surface elements.*



Everything you select in *Edit* mode will get a white and red-colored frame.

In order to edit the surface name you have to select it and change the name with the *Text tool* '**A**' from the *Tool Bar*. The same is possible for all text fields on the surface.

To change the colors you have to open the *Color Selector* from the *Tool Bar*.

Now, as an example, we are going to change the color of the panel.

If you click on the frame of the panel, the mouse pointer will change into a blue cross and you can move the selected element around in the *Project Window.* If you click on a side of the white and red-colored frame the mouse pointer will change into an arrow and you can resize the item.

The *Color Selector* shows the current color of the selected panel. In order to change the color you have to scroll around the color circle with the pressed left mouse button, dragging the point with the mouse.

To get the nuances of the selected color you can move around in the color

box and click on the right nuance. You see, that the color of the panel updates interactively into this color.

You can repeat this procedure with the texts of the surface and other surface elements. Just experiment a little bit and try some of the various options.
It is a good idea to save the project now.

We have already created a new module with its adapted surface but there is no option to turn off signal A. In the following step we want to extend our circuit with this extra functionality. However we still want do make use of the release fade that is generated by the ENVELOPE. An on/off switch that is inserted into the Gate connection would do the trick.

Index

A

ADAT 8
Adjusting Surface Controls 21
Adjusting the I/O settings 8
ADSR EG 16
alias 4
Attack 17
Audio Trigger 15, 18

B

basic modules 13
button 13

C

circuit 19
Circuit Design 9, 14
Circuit view 24
Clock 8
Close 5, 10
Color Selector 24
command bar 6
context menu 23
Curve 22

D

Designing 23
desktop 4
Divisor 22

E

Edit 24
Effects 14
Enter 13
Envelope Follower 14, 17

F

F1 15
F11 20
F12 19
Fader 14
File Browser 5, 9, 14
Fold button 19
Follow 18

G

Gain 17, 18
Gates 15

H

Help Window 14, 15

I

I/O settings 8
I/Os 6
input 6
input indicator 14
Input/Output Gain 17
Intensity 22
interfaces 13

K

Keep on Top 10

L

Layout 23
library 9

M

Max 22
Menu Bar 5
Min 22
Minimize 5
Mix 1 18
Modulation 14
module 9
Move 24

N

Nodes 6

O

Open Panel 14
Oscillators 10
Output 8
outputs 6

P

Pad 11, 20
Pad List window 21
Pads 10
parameters 13
Project Window 5, 6, 14, 20
ProjectExplorer) 20

R

Release 17
roject Window 9

S

Sample Rate 8
Sample Rate Settings 8
Save as 12
SCOPE board 6
SCOPE icon 4
SCOPE library 9
Scripting 25
Sensitivity 18
Setting up SCOPE 4
Settings 4
Signal Adders/Mix 10
Sine Generator 17
slider 13
Source 18
Surface Controls 21
Surface view 23, 24
Surfaces 14

T

Test Generator 17
Text tool 19, 24
Threshold 18
title 6
Tool Bar 5, 15, 19
Tools 14

U

Use 24

V

Vars 22
View 4, 8

W

workflow 6