The dNα-Files

SDK Usage tips – General Tips

SDK Development for
the Sonic Core Scope Platform

dNa – Digital&Analog

DSP Plugins for Sonic Core Scope platform

Preface:

Thank you for reading upon my Sonic Core Scope SDK development tips and tricks.
I made the screenshots and examples in SDK 4 but that should not be a problem,
since the latest v7 is built from this version as I understood from Sonic Core.

I had planned for a long time now to give some tips and tricks and good habits for the development on Scope. In this reader I will list some general tips and tricks.

With dNa, I have come a long way in developing plugins for the Scope Platform; main goal was always trying to create essential and intuitive plugins, which combine the best of my analog and digital experiences into the dNa products. I learned the SDK by a lot of trial and error. This reader will hopefully contribute to maintaining the high standard that is called SCOPE.

One thing is for sure, I am very grateful for the users supporting dNa and Sonic Core in keeping Scope alive.

A special thanks goes out to the PlanetZ, Hitfoundry, OSS and FB friends and supporters with whom I've always had very nice personal contact with. Sorry i won't name you all, since I couldn't forgive myself if I forgot one of you, you know who you are! But one person in particular I have to name: Holger for making and keeping it all possible after all these years, and for being a good friend.

Sincerely,

Ray de Jager
dNa – Digital&Analog

General:

- Scope SDK is known also for having a lot of undocumented features. In all the windows check if there are any by right clicking on something and see if there are hidden menus or options. This can be done on pads, folders and different types of modules.
- Save your progress...not just as project...since those can be corrupted...but the module itself you are developing. I use 1_module x.mdl, 2_module x.mdl, 3_module x.mdl and so forth. Make a habit out of this! Cannot emphasize this enough!!!
- Make good use of 'fold' for folding circuits and 'group' for grouping GO/gui stuff. This keeps everything organized for yourself
- A good tip for beginners....don't just connect your gui stuff directly to atoms or the basic modules, but use a dummy pad for that. This way if you change something internally(or parameterize) later on your connections are not lost. Try to make most of those connectors...try understanding pads, pad types and vars.
- Use drag and drop to connect pads in the two pad windows. Much faster than connect button.

GUI:

- The viewtree group options(GO Tree Editor) are important while developing a GUI(auto resizes a group) and setting to fixed custom sizes at final stage for fast and smooth gui. The info is sparse but trial and error and maybe some guidance will get you there.
- Making tooltips: rename your fader, knob or button with a '@' prefix. For example you have a pot controlling a gain module...instead of naming it 'gain' name it '@gain'
- On a GUI, next to the tooltip, db values could be displayed from the underlaying gain module. If you want to prevent this use a diode in between. A diode only gives values from in to out. Not bidirectional.
- Use .tga files 32 bit for graphics (for example Photoshop)

DSP:

- after figuring out your circuit...ask yourself "can I build it sleeker?...with less dsp modules giving the same result?" Sometimes it is a good habit to rethink what you did. Dsp is limited on pci.
- in sdk/scope certain values and flags can be set back using the value '-1' for 'not set'
- never set/assign dsp atoms or circuits to dedicated dsp chips! Keep it compatible and dynamic as possible.
- Instead figure out how to use the 'dynvoicesofparent' module to make things very dsp efficient. A switch combined with an 'if'-module is very powerful. Use '0' to unload from dsp and '-1' to set it back to not set(loads on dsp by default) This is where basic programming skills are handy since your essentially building an 'if then else'-line

Finalizing:

- Before protecting and exporting delete circuit go's and optimize internally. Before you do this....SAVE IT for crying out loud! Protected means closed forever! There is no undo!
- If you finished a device and exported it...rename the mdl to a .dev

On certain subjects I will publish dedicated readers.