

Holy City Audio Forum

Former Home of SpinCAD Designer

[Skip to content](#)

Search...

[Advanced search](#)

[Moderator Control Panel]

SpinCAD Builder - the Volume block

Post a reply

Search this topic...

2 posts • Page 1 of 1

- [Edit post](#) (./posting.php?mode=edit&f=42&p=2530)
- [Delete post](#) (./posting.php?mode=delete&f=42&p=2530)
- [Report this post](#) (./report.php?f=42&p=2530)
- [Warn user](#) (./mcp.php?i=warn&mode=warn_post&f=42&p=2530&sid=d6d1dcc4501596550704606791e62b4c)
- [Information](#) (./mcp.php?i=main&mode=post_details&f=42&p=2530&sid=d6d1dcc4501596550704606791e62b4c)
- [Reply with quote](#) (./posting.php?mode=quote&f=42&p=2530)

SpinCAD Builder - the Volume block (#p2530)

by **drolo** » Fri Oct 30, 2015 8:48 am

I would certainly be interested in making blocks but must say that I have really little experience with programing. SpinASM is the first programming I have ever done ... so before you invest time into putting something together check if there are other people interested in it because it just might be wasted on me.

I do this for myself so I'm not too concerned about sharing anything. For commercial builders I guess DSP code has become the last way to create something that is not "clonable" which might explain their reluctance to share.

Top

- [Edit post](#) (./posting.php?mode=edit&f=42&p=2531)
- [Delete post](#) (./posting.php?mode=delete&f=42&p=2531)
- [Report this post](#) (./report.php?f=42&p=2531)
- [Information](#) (./mcp.php?i=main&mode=post_details&f=42&p=2531&sid=d6d1dcc4501596550704606791e62b4c)
- [Reply with quote](#) (./posting.php?mode=quote&f=42&p=2531)

Re: SpinCAD Designer Bleeding Edge Experimental 0.98 build 9 (#p2531)

by **Digital Larry** » Fri Oct 30, 2015 9:20 am

The whole idea about SpinCAD Builder is that you just need to add a little "icing" to your Spin ASM code to allow it to become a block. Here's the SpinCAD Builder code for the volume block, for example:

<https://github.com/HolyCityAudio/SpinCAD-Designer/blob/master/src/SpinCADBuilder/Volume.spincad>

```
@name Volume
@color "0x2468f2"
@audioInput input Input
@audioOutput output Output
@controlInput volume Volume

equ gain 1.0
@sliderLabel gain 'Input Gain' -24 0 -6 1.0 1 DBLEVEL

;REGISTERS
equ output reg0

@isPinConnected Input
rdax input, gain ;read input signal
@isPinConnected Volume
mulx volume
@endif

wrax output,0 ;output in ACC and lf1

@setOutputPin Output output
@endif
```

Bit by bit:

```
@name Volume
the name of this block is Volume
```

@color "0x2468f2"
sets the color for this block

@audioInput input Input
define an audio input pin. these appear on the top of the block. Its name is "Input" (as shown when hovering the mouse over a pin). It is represented internally by the variable "input". Pin variables ultimately resolve to register values.

@audioOutput output Output
define an audio output pin. these appear on the bottom of the block. Its name is "Output". It is represented internally by the variable "output".

@controlInput volume Volume
define an control input pin. these appear on the left side of the block. Its name is "Volume". It is represented internally by the variable "volume".

equ gain 1.0
@sliderLabel gain 'Input Gain' -24 0 -6 1.0 1 DBLEVEL

Specify an internal variable called "gain". Define a corresponding slider in the control panel that will adjust this variable. The slider will be called "Input Gain". Its range will be from -24 to 0 dB, default setting -6 dB. There is one digit to the right of the decimal point in the displayed value. The mapping of slider setting to the actual gain variable will be according to the "DBLEVEL" function. So if you set the slider to -6, gain is actually 0.5.

;REGISTERS
equ output reg0
Define registers that are needed by this algorithm. Output values always need to have a register defined.

@isPinConnected Input
This macro checks to see whether the input pin is connected in this model. You have to find the matching "@endif" which in this case is at the end. What this means is: If the input pin is not connected, don't generate any code for this block". More generally, it allows setting default values if control inputs are not connected.

rdax input, gain ;read input signal
Here's your actual Spin ASM instruction, literally

@isPinConnected Volume
mulx volume
@endif

Here's where we say "if the Volume pin is connected, multiply the scaled input value by that register". Note that "volume" could resolve to a pot value or any computed value in your program. If the volume pin is NOT connected, we do not generate the mulx instruction.

wrax output,0 ;output in ACC
Finally, write the result to the output register.

@setOutputPin Output output
Here's where we tell SpinCAD that there's an Output Pin whose name is "Output" and is represented by the register variable "output". In some ways this seems redundant with the @audioOutput statement. I can't remember whether I decided if it was possible to remove this. In any case, it's no big deal.

@endif
This is the @endif that matches the @isPinConnected at the beginning of the program.

That's your intro to SpinCAD Builder language.

Top

Display posts from previous: All posts Sort by Post time Ascending Go

Post a reply

2 posts • Page 1 of 1

Return to Developer's Corner

Jump to: Developer's Corner Go

Quick-mod tools: Lock topic Go

Who is online

Users browsing this forum: Digital Larry and 0 guests
Powered by phpBB® Forum Software © phpBB Group

Administration Control Panel