# Holy City Audio Forum

Former Home of SpinCAD Designer



## All-Pass Filter Block

Post a reply



### All-Pass Filter Block (#p1567)

by **disasterarea** » Thu May 22, 2014 3:28 pm

I would love to have a modular all-pass filter with a control input. We have the phaser block but there's zero flexibility there.

If we just have the ability to input the coefficient, then we could chain up a bunch of them and then just patch the control inputs together.

This is from the Spin forum - mdroberts1243.

```
wrax temp, 0
rda delay#, 1 ; allpass with WRAP replaced to use variable coefficient
mulx coefficient
rdax temp,1 ; add input *NEW*
wra delay, -1 ; store to delay and then negate ACC
mulx coefficient ; apply 'negative' coefficient
rda delay#, 1 ; add in the end entry in the delay
```

Top

- Edit post (./posting.php?mode=edit&f=31&p=1568)
- Delete post (./posting.php?mode=delete&f=31&p=1568)
- Report this post (./report.php?f=31&p=1568)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1568&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1568)

### Re: All-Pass Filter Block (#p1568)

by **Digital Larry** » Thu May 22, 2014 4:02 pm

Thanks for the input. Looks easy. Of course I still need to get my development environment working again... maybe I'll spend some time on that tonight.

Top
- Edit post (./posting.php?mode=edit&f=31&p=1743)
- Delete post (./posting.php?mode=delete&f=31&p=1743)
- Report this post (./report.php?f=31&p=1743)
- Warn user (./mcp.php?i=warn&mode=warn_post&f=31&p=1743&sid=befaf904784df501ab581b011c284453)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1743&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1743)

## Re: All-Pass Filter Block (#p1743)

by **disasterarea** » Thu Aug 14, 2014 10:25 am

Hi, Larry - any progress on the all-pass block? The phaser works now in 759, thanks for fixing it, but you can't chain any of them together and you're stuck with an 8-stage phaser. It's also not possible to tweak the LFO rate or etc. Having a dedicated block so we could roll our own would be great.

Thanks!

Top
- Edit post (./posting.php?mode=edit&f=31&p=1744)
- Delete post (./posting.php?mode=delete&f=31&p=1744)
- Report this post (./report.php?f=31&p=1744)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1744&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1744)

## Re: All-Pass Filter Block (#p1744)

by **Digital Larry** » Thu Aug 14, 2014 6:51 pm

Hey - on vacation with limited Wi-Fi, but thanks for reminding me, I'll see what I can pull together for you. Probably early next week before I can get a new release posted. Other than fixing the, what was it, control smoother block, I added a batch convert (SPCD to SPN) function and also implemented the pitch "offset" function which I found at the Spin Knowledge Base - it's pretty special purpose (cough).

OK I just copied it to my magical translator and the following questions come to mind:

What range do you want "delay" to cover?
Where is "coefficient" going to come from? As you have laid it out, it is a register value. I'm not sure if you wish this to be a control input or settable via control panel, or adjustable via control input, but with a default value when no input is connected.

Thx,,DL

Top
- Edit post (./posting.php?mode=edit&f=31&p=1746)
- Delete post (./posting.php?mode=delete&f=31&p=1746)
- Report this post (./report.php?f=31&p=1746)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1746&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1746)

## Re: All-Pass Filter Block (#p1746)

by **Digital Larry** » Fri Aug 15, 2014 7:38 am

So far, here is what I have in "SpinCAD Builder" language. This is an extension of Spin ASM which allows easy creation of blocks for the SpinCAD program. I'm still working on getting control panels to generate from the same source... lotsa work, as usual...

```
@name Allpass
@audioInput input Input
@audioOutput output1 Output

mem delay 1
equ temp reg0
equ coefficient reg1
equ output1 reg2

rdax input, 1.0
wrax temp, 0
rda delay#, 1 ; allpass with WRAP replaced to use variable coefficient
mulx coefficient
rdax temp,1 ; add input *NEW*
wra delay, -1 ; store to delay and then negate ACC
mulx coefficient ; apply 'negative' coefficient
rda delay#, 1 ; add in the end entry in the delay
wrax output1, 0.0

@setOutputPin Output output1
```

SpinCAD audio Blocks require something to come in as the input, and they also dedicate a register to hold the block's output result. I believe that the syntax is fairly obvious, but since I wrote it, that's to be expected.

Top

- Edit post (./posting.php?mode=edit&f=31&p=1747)
- Delete post (./posting.php?mode=delete&f=31&p=1747)
- Report this post (./report.php?f=31&p=1747)
- Warn user (./mcp.php?i=warn&mode=warn_post&f=31&p=1747&sid=befaf904784df501ab581b011c284453)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1747&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1747)

## Re: All-Pass Filter Block (#p1747)

by **disasterarea** » Fri Aug 15, 2014 7:50 am

Larry, the "delay" is just a single block to hold the audio. "Coefficient" is the "K" value, the all-pass coefficient. Ideally the all pass block would just have one control input - the "coefficient" / K input. You'd need to feed this with a pot or LFO value, which we'd wire up with other blocks.

The current Phaser block implements essentially the same thing, in an 8-stage design:

```
;  Program: Render Block exported from SpinCAD Designer

;------ Input
;------ Mixer 2-1
RDAX ADCL,0.5000000000
WRAX REG0,0.0000000000
RDAX ADCR,0.5000000000
RDAX REG0,1.0000000000
WRAX REG0,0.0000000000

;------ Feedback Output
;------ Pot 0
;------ Mixer 2-1
RDAX REG0,0.5000000000
WRAX REG2,0.0000000000
RDAX REG1,0.5000000000
MULX POT0
RDAX REG2,1.0000000000
WRAX REG2,0.0000000000

;------ Pot 1
;------ Pot 2
;------ Phaser
SKP RUN ,1
WLDS 1,0,32767

RDAX POT1,1.0000000000
RDAX REG15,0.9000000000
WRAX REG15,0.0000000000
RDAX POT2,1.0000000000
MULX POT2
SOF 0.2000000000,0.0200000000
WRAX SIN1_RATE,0.0000000000
CHO RDAL,1
SOF 0.5000000000,0.5000000000
LOG 0.5,0.0
EXP 1.0,0.0
SOF 1.0000000000,-0.5000000000
SOF 1.9990000000,0.0000000000
MULX POT1
SOF 0.1000000000,0.8500000000

WRAX REG3,0.0000000000
RDAX REG5,1.0000000000
WRAX REG13,1.0000000000
MULX REG3
RDAX REG2,0.0156250000
```

```
WRAX REG5,-1.0000000000
MULX REG3
RDAX REG13,1.0000000000
WRAX REG14,0.0000000000
RDAX REG6,1.0000000000

WRAX REG13,1.0000000000
MULX REG3
RDAX REG14,1.0000000000
WRAX REG6,-1.0000000000
MULX REG3
RDAX REG13,1.0000000000
WRAX REG14,0.0000000000
RDAX REG7,1.0000000000

WRAX REG13,1.0000000000
MULX REG3
RDAX REG14,1.0000000000
WRAX REG7,-1.0000000000
MULX REG3
RDAX REG13,1.0000000000
WRAX REG14,0.0000000000
RDAX REG8,1.0000000000

WRAX REG13,1.0000000000
MULX REG3
RDAX REG14,1.0000000000
WRAX REG8,-1.0000000000
MULX REG3
RDAX REG13,1.0000000000
WRAX REG14,0.0000000000
RDAX REG9,1.0000000000

WRAX REG13,1.0000000000
MULX REG3
RDAX REG14,1.0000000000
WRAX REG9,-1.0000000000
MULX REG3
RDAX REG13,1.0000000000
WRAX REG14,0.0000000000
RDAX REG10,1.0000000000

WRAX REG13,1.0000000000
MULX REG3
RDAX REG14,1.0000000000
WRAX REG10,-1.0000000000
MULX REG3
```

```
RDAX REG13,1.0000000000
WRAX REG14,0.0000000000
RDAX REG11,1.0000000000

WRAX REG13,1.0000000000
MULX REG3
RDAX REG14,1.0000000000
WRAX REG11,-1.0000000000
MULX REG3
RDAX REG13,1.0000000000
WRAX REG14,0.0000000000
RDAX REG12,1.0000000000

WRAX REG13,1.0000000000
MULX REG3
RDAX REG14,1.0000000000
WRAX REG12,-1.0000000000
MULX REG3
RDAX REG13,1.0000000000

SOF -2.0000000000,0.0000000000
SOF -2.0000000000,0.0000000000
SOF -2.0000000000,0.0000000000
SOF -2.0000000000,0.0000000000
SOF -2.0000000000,0.0000000000
SOF -2.0000000000,0.0000000000
MULX REG15
RDAX REG2,1.0000000000
WRAX REG4,1.0000000000
;------ FB In 1
RDAX REG4,1.0000000000
WRAX REG1,0.0000000000
;------ Output
RDAX REG4,1.0000000000
WRAX DACL,0.0000000000
RDAX REG0,1.0000000000
WRAX DACR,0.0000000000
```

The difference as I see it is that the example code is using up a register at each pass (REG5-REG12,) if you were doing this by hand there would be no reason not to use the same register for each stage. It's got a fixed 8-stage topology and it's doing some stuff with the LFO - the LOG and EXP statements appear to be shaping the wave driving the all-passes.

I'd like to be able to do neat stuff like set up lots of phaser stages and then use a pot...skp routine to turn some of them off, or change the phase coefficients for each stage separately using a SOF statement between the LFO and the control input. Smaller blocks are always better for tweakers like me!

Top
- Edit post (./posting.php?mode=edit&f=31&p=1749)
- Delete post (./posting.php?mode=delete&f=31&p=1749)
- Report this post (./report.php?f=31&p=1749)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1749&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1749)

## Re: All-Pass Filter Block (#p1749)

by **Digital Larry** » Sat Aug 16, 2014 5:35 am

I spent some time last night and got this block "basically" working. There are a few odd things about it and so I will disclaim it as "experimental" when I pull the release together.

#1 As you might notice from the Spin-supplied phaser code, you really have to scale the input to the all pass stages DOWN or else you get massive distortion. Then at the end you need to scale it way UP again (though SpinCAD currently lacks a gain boost block). I tried to figure out why this would happen, and I'm sure there's a mathematical explanation - but it's eluding me.

#2 The implementation using memory instead of registers helps not eat so many registers, which is offset by the fact the SpinCAD wants to use a register at the output of every block whether it's needed or not. But those can be hand optimized.

#3 Something about the memory allocation routines that ElmGen uses prevents allocation of a single memory location - you always get 2. So, if I allocate 1 memory location for **delay** then **delay** would be 0, but **delay#** will be 1. I could get around this by simply referring to **delay** in both cases, and the extra few memory locations wasted aren't going to hurt anything.

#4 It looks like the control signal processing is pretty important. I tried putting in a 0 - 1 amplitude sine wave and it definitely gets some unpleasant *BEEYOWR* at the extremes of the sweep. Using a control smoother or scale/offset can help, but the phase effect is pretty subtle then. However I didn't try my own suggestion in point #3 yet, which *may* be reducing the phase shift produced per stage.

As with many things, some experimentation is in order, and it may also involve some hand coding to know what works best. I have no doubt that the control processing in Spin's phaser example was the result of a fair amount of experimentation.

Top
- Edit post (./posting.php?mode=edit&f=31&p=1750)
- Delete post (./posting.php?mode=delete&f=31&p=1750)
- Report this post (./report.php?f=31&p=1750)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1750&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1750)

## Re: All-Pass Filter Block (#p1750)

by **Digital Larry** » Sat Aug 16, 2014 7:16 am

Another possibility we may wish to explore is simply adding the desired flexibility to the phaser block itself, such as # of stages, alternate control inputs, etc. As you probably already guessed, SpinCAD doesn't support pot skipping structures, and it would not be that easy to add it (not impossible, just requires some architectural enhancements that I wasn't planning short term).

Top

- Edit post (./posting.php?mode=edit&f=31&p=1752)
- Delete post (./posting.php?mode=delete&f=31&p=1752)
- Report this post (./report.php?f=31&p=1752)
- Warn user (./mcp.php?i=warn&mode=warn_post&f=31&p=1752&sid=befaf904784df501ab581b011c284453)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1752&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1752)

## Re: All-Pass Filter Block (#p1752)

by **disasterarea** » Thu Aug 21, 2014 7:00 pm

Just being able to choose the number of phase stages would be hugely helpful. I'd say a maximum of 16 stages if we have enough instructions available, but to do that we might need to re-use the same register / delay location. Looking at the example code, one thing that jumps out at me is that the "phase" variable is controlling both width and mix!

The example is reverb + phase, which is fine. Here's the phase shifter block, it's pretty much identical to the code generated by SPCD:

```
;Do phase shifter from sin1:

rdax   pot2,1
rdax   bypass,0.9
wrax   bypass,0

rdax   pot1,1
mulx   pot1
sof   0.2,0.02
wrax   sin1_rate,0

cho   rdal,sin1      ;read sin1 as +/-1
sof   0.5,0.5      ;make positive only sin ranges 0 to 1
log   0.5,0
exp   1,0      ;square root function
sof   1,-0.5      ;make +/-0.5
sof   1.999,0      ;make +/-1 again
mulx   pot2      ;pot2 controls width and mix
sof   0.1,0.85
wrax   phase,0      ;phase variable ranges 0.8 to 0.95

rdax   p1,1
wrax   temp,1
mulx   phase
rdax   mono,1/64   ;input to phase shift network
wrax   p1,-1
mulx   phase
rdax   temp,1
```

```
wrax   temp1,0

rdax   p2,1
wrax   temp,1
mulx   phase
rdax   temp1,1
wrax   p2,-1
mulx   phase
rdax   temp,1
wrax   temp1,0

rdax   p3,1
wrax   temp,1
mulx   phase
rdax   temp1,1
wrax   p3,-1
mulx   phase
rdax   temp,1
wrax   temp1,0

rdax   p4,1
wrax   temp,1
mulx   phase
rdax   temp1,1
wrax   p4,-1
mulx   phase
rdax   temp,1
wrax   temp1,0
rdax   p5,1
wrax   temp,1

mulx   phase
rdax   temp1,1
wrax   p5,-1
mulx   phase
rdax   temp,1
wrax   temp1,0

rdax   p6,1
wrax   temp,1
mulx   phase
rdax   temp1,1
wrax   p6,-1
mulx   phase
rdax   temp,1

sof   -2,0
```

```
sof   -2,0
sof   -2,0
sof   -2,0
sof   -2,0
sof   -2,0   ;output of phase shifter in acc

mulx   bypass
rdax   mono,1
wrax   pout,1
rdax   revout,1
sof   1,0.02
wrax   dacl,1
sof   1,-0.04
wrax   dacr,0
```

It's a bit different than your AP filter code - you're doing rdax input, 1; wrax temp, 0. The example is rdax p1,1 (same) then wrax temp, 1, then mulx phase. You're reading in the delay block there and I think it might be unnecessary. The rdax mono, 1/64 is the input to the first filter. So that's where it's scaling down, and then there are 6 sof -2.0 blocks at the end of the chain. Divide by 64, then multiply by 2^6 = 64. Unity gain through the chain, but if we can choose the number of stages then we have to adjust the input and output gain to match or things will get weird.

The other part to notice are the comments on the LFO portion - the phase constant ranges between 0.8 and 0.95. Lower than that and the mix is super low, and getting all the way to 1.0 probably results in positive feedback if you have enough stages in line. I've been experimenting with adding a feedback loop and values over 0.5 on the control yield self-oscillation. Wheeee-oooooh-wheeeee-oooooh! Fun but I'll take it out.

So since there are a ton of weird things that have to happen in order to control the phaser, I'm inclined to agree that we just need a more flexible phaser instead of a modular block. Boo.

So here's what we know about the phaser:

The two mulx operations per block control phase width and mix. It would be worth figuring out which is which, I'll try tweaking the code by hand tomorrow and report back. Once we know how to control them, it should be pretty simple to assign a slider and input for each one separately.

The phase coefficient that is used for the mulx's needs to range between 0.8 and 0.95.

Keith Barr apparently felt that the waveshape needed to be modified quite a bit (positive only, log, exp, then level shifting.) That's probably hyper-triangular or similar, spending more time in the "interesting" parts of the cycle. Slacker has a badass flanger algorithm that I've been meaning to monkey with, and he uses absa and some sof's to make the same kind of shape. This one is probably hyperbolic, now that I come to think about it.

How feasible do you think a more controllable phaser would be? Thinking we'd need to be able to select stages by twos with correctly scaling makeup gain in front and behind. We'd also need

controls for width and mix, and an input for an LFO. I really prefer not to have the LFO stuff built in, because then you lose the ability to drive the thing manually.

I've been doing some thinking about the number of stages - 6 stages requires a gain of 1/64 to 64x. I'm thinking each stage needs to drop the gain by two. I don't know how far we can take everything before it all falls apart - 8? 12? 16?

Top
- Edit post (./posting.php?mode=edit&f=31&p=1753)
- Delete post (./posting.php?mode=delete&f=31&p=1753)
- Report this post (./report.php?f=31&p=1753)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1753&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1753)

## Re: All-Pass Filter Block (#p1753)

by **Digital Larry** » Thu Aug 21, 2014 8:11 pm

OK, first of all I have to say that I really just have to crank out this release that has the fix for the control smoother and the other few things. Continuing to work on stuff usually rips it apart so it's worse for awhile before it gets better. So I'm not going to fix the pitch calculation for semitones quite yet because now we're talking phaser.

I love phaser so there's no problem there. Let me get this other thing done and then we work on phaser.

I was able to get 10 blocks chained together and it did sorta "phase".

I can pull the Phaser LFO thing "off" and make it its own block separate from the all-pass chain itself.

You can put a slider control on any constant in your algorithm, either directly or through some mathematical transformation. It can also control block structure, as in the # of stages or filter poles. Since it's written in Java, at this stage it's very flexible.

So yes, I think having a control panel slider (as in the Overdrive), radio button (hate em), or drop down (OK) to select the number of stages is fine. And good thing you noticed the scale down/up relationship; seems to make sense this can easily be tied in with the # of stages.

I'm looking at the code and scratching my head. I thought I copied what you supplied and just made it into a block. Will have to look at it later.

Trying not to get distracted...

Thanks!!!

Top
- Edit post (./posting.php?mode=edit&f=31&p=1755)
- Delete post (./posting.php?mode=delete&f=31&p=1755)
- Report this post (./report.php?f=31&p=1755)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1755&sid=befaf904784df501ab581b011c284453)

- Reply with quote (./posting.php?mode=quote&f=31&p=1755)

## Re: All-Pass Filter Block (#p1755)

by **Digital Larry** » Fri Aug 22, 2014 12:15 pm

Ok I put up the bug fix release #823 last night, so now I can concentrate a bit more on the phaser!

I'm at a loss to explain the differences between what you originally posted and the Spin phaser example. I feel better when I can work out the math on these things but I haven't yet sat down to try. I think if I study this for several years, I may get it.

https://ccrma.stanford.edu/~jos/pasp/Al ... Combs.html (https://ccrma.stanford.edu/~jos/pasp/Allpass_Two_Combs.html)

Note this comment:

> This can be recognized as direct form I [452], which requires 2M delays instead of M ; however, unlike direct-form II, direct-form I cannot suffer from ``internal'' overflow--overflow can happen only at the output.

The combined "width/mix" business is a little odd. With my delay blocks I have attempted to keep them 100% wet - then you can mix with the dry signal whichever way you like. If "mix" is only variable between 0.8 and 0.95, it's not making that much of a difference anyway. I'd like to try to have a 100% wet phase shift/allpass stage. With no dry signal this could give you a vibrato effect.

I'm also a bit curious about using delay vs. registers and the scaling down/back up issue. As you know, the delay RAM is floating-point compressed, so to speak, so I would expect somehow that scaling signals way down then way up in RAM would lead to a noisier result than using registers for the delay elements. We can experiment both ways. I also wonder, somewhat along those lines, whether it makes more sense to recover gain via **SOF -2.0, 0** after every pair of phase shift elements, or just do it all BAM at the end. Again it can be tried both ways. At this point I still don't know WHY the scaling is required, other than if you try to skip it, bad things do happen.

Top

- Edit post (./posting.php?mode=edit&f=31&p=1757)
- Delete post (./posting.php?mode=delete&f=31&p=1757)
- Report this post (./report.php?f=31&p=1757)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1757&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1757)

## Re: All-Pass Filter Block (#p1757)

by **Digital Larry** » Sat Aug 23, 2014 8:03 pm

I made a stand-alone block from the "square root" part of the phaser's control signal processing, then ran an LFO going from 0 to 1 through it. Here's the result:

Image (https://imageshack.com/i/p996EtlZp)

Sine wave is magenta. Square root is light blue. Vertical scale is logarithmic.

Top
- Edit post (./posting.php?mode=edit&f=31&p=1758)
- Delete post (./posting.php?mode=delete&f=31&p=1758)
- Report this post (./report.php?f=31&p=1758)
- Warn user (./mcp.php?i=warn&mode=warn_post&f=31&p=1758&sid=befaf904784df501ab581b011c284453)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1758&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1758)

## Re: All-Pass Filter Block (#p1758)

by **disasterarea** » Wed Aug 27, 2014 8:42 am

I like the idea of being able to process other control signals with the sqrt block! Could come in VERY handy!

I'll be working with 823 over the next few days, looking forward to seeing what you can do on the phaser.

Top
- Edit post (./posting.php?mode=edit&f=31&p=1767)
- Delete post (./posting.php?mode=delete&f=31&p=1767)
- Report this post (./report.php?f=31&p=1767)
- Information (./mcp.php?i=main&mode=post_details&f=31&p=1767&sid=befaf904784df501ab581b011c284453)
- Reply with quote (./posting.php?mode=quote&f=31&p=1767)

## Re: All-Pass Filter Block (#p1767)

by **Digital Larry** » Wed Sep 03, 2014 11:34 am

I've done a bit more work and a lot more thinking, too bad it wasn't the other way around!

Anyway, what shows up in the Spin phaser example as the "mix" or "depth" (can't remember), where you multiply one of the intermediate results by the phase control input again, is not actually a "depth" or "mix" control. It is needed to keep the gain of the all pass filter at 1, and has to be left in.

Currently the experimental build 837 has all phase control inputs connected to a single input point. I can easily bring them out to individual control inputs, I'm just wondering the best way to do this. My initial thought would be to have a control panel selector with the following options:

a) Built-in LFO
b) Single phase control input
c) Individual phase control inputs

I've also made the "Root" function block more flexible, in that it can now do roots 2 through 5. It could certainly go beyond or get in between the existing points, but I'm waiting for proof that this flexibility is actually needed. I think that an inverted power function might be the best for phaser control. Last night I checked out Slacker's phaser patch from his "Babelfish" project and it sounds very nice compared to the Spin example. I'm also putting more attention into making sure the control processors such as power and clip work correctly. I found at least two serious bugs in the power control that would have prevented it from being used properly under some conditions.

Top

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

Post a reply

14 posts • Page **1** of **1**

Return to SpinCADBlocks

Jump to: SpinCADBlocks Go

Quick-mod tools: Lock topic Go

## Who is online

Users browsing this forum: Digital Larry and 0 guests



**Administration Control Panel**