

Holy City Audio Forum

Former Home of SpinCAD Designer

[Skip to content](#)

Search...

[Advanced search](#)

[[Moderator Control Panel](#)]

Hand optimizing Spin ASM

Post a reply

Search this topic...

4 posts • Page 1 of 1

- [Edit post \(./posting.php?mode=edit&f=32&p=1781\)](#)
- [Delete post \(./posting.php?mode=delete&f=32&p=1781\)](#)
- [Report this post \(./report.php?f=32&p=1781\)](#)
- [Information \(./mcp.php?i=main&mode=post_details&f=32&p=1781&sid=90615c931909e0633cbcd2a3573379b1\)](#)
- [Reply with quote \(./posting.php?mode=quote&f=32&p=1781\)](#)

Hand optimizing Spin ASM (#p1781)

by **Digital Larry** » Sun Sep 14, 2014 6:42 am

If you've looked at Spin ASM code created by SpinCAD Designer, you may have noticed that it sometimes does things it doesn't have to. For example, sometimes it stores a value into a register and clears the accumulator, only to read that value out of the register and multiply it by 1.0 in the next step. Maybe I'll explain why this happens later, but for now I will explain what you can do about it.

Here's a sample patch that we'll work on. It is small and simple, just to keep the discussion focused.

[Image \(https://imageshack.com/i/eyLp8AKup\)](https://imageshack.com/i/eyLp8AKup)

Let's understand the first rule of optimization:

Don't optimize unless your program length is over 128 instructions! Why not? There's simply no benefit to it. If your program is under 128 instructions already, you don't get any brownie points for making it a few instructions shorter. Now, add a block until it IS over 128 - NOW you can try optimizing.

OK let's move on to identify areas which can easily be optimized.

Example 1: Final block output goes to both left and right audio output, with a gain of 1.0 on both sides.

[Image \(https://imageshack.com/i/kmmaVhZOp\)](https://imageshack.com/i/kmmaVhZOp)

Here's the before code in this area:

```
; Program: Render Block exported from SpinCAD Designer
.... (statements at the beginning removed)
MULX REG1
RDAX REG3,1.0000000000
WRAX REG3,0.0000000000
;----- Output
RDAX REG5,1.0000000000
WRAX DACL,0.0000000000 ; change gain from 0.0 to 1.0
RDAX REG5,1.0000000000 ; remove this instruction
WRAX DACR,0.0000000000
```

Here's the after code:

```
; Program: Render Block exported from SpinCAD Designer
.... (statements at the beginning removed)
MULX REG1
RDAX REG3,1.0000000000
WRAX REG3,0.0000000000
;----- Output
RDAX REG5,1.0000000000
WRAX DACL,1.0000000000
WRAX DACR,0.0000000000
```

1 instruction saved! OK it's not much, but sometimes it's all you need.

Another thing to consider, is that if your pedal design is mono out, then there's no point to write the result to both outputs. If, in the previous example, you are really only using the left output, then change it like this:

Before:

```
;----- Output
RDAX REG5,1.0000000000
WRAX DACL,1.0000000000
WRAX DACR,0.0000000000
```

After:

```
;----- Output
RDAX REG5,1.0000000000
WRAX DACL,0.0000000000 ; gain should be 0.0 here
```

1 more instruction saved! In this case, the easiest thing to do would be to delete the connection to the Output Block's Right Input in your SpinCAD Design.

Example 2: Destination block follows source block and is the only destination.

In this case you will see this pattern:

```
WRAX REGx,0.0000000000 ; you can remove this instruction
;----- Next Block
RDAX REGx,1.0000000000 ; and you can remove this instruction
```

The previous block puts its result into a register. Then the following block reads that value out of the same register and proceeds to do something with it. As long as the instructions are back to back, (other than comments), it is safe to remove both of these instructions.

```
; Program: Render Block exported from SpinCAD Designer
;----- Input
;----- Cube gain
RDAX ADCL,1.0000000000
WRAX REG0,-0.9333300000
MULX REG0
MULX REG0
RDAX REG0,1.0000000000
SOF 1.5000000000,0.0000000000
WRAX REG1,0.0000000000 ; you can remove this instruction
;----- Output
RDAX REG1,1.0000000000 ; and you can remove this instruction
WRAX DACL,0.0000000000
```

```
; Program: Render Block exported from SpinCAD Designer
;----- Input
;----- Cube gain
RDAX ADCL,1.0000000000
WRAX REG0,-0.9333300000
MULX REG0
MULX REG0
RDAX REG0,1.0000000000
SOF 1.5000000000,0.0000000000
;----- Output
WRAX DACL,0.0000000000
```

2 instructions saved!

Make sure that REGx is not used any further on in the algorithm. If it is, you can still do this to save one instruction:

```
; Program: Render Block exported from SpinCAD Designer
;----- Input
;----- Cube gain
RDAX ADCL,1.0000000000
WRAX REG0,-0.9333300000
MULX REG0
```

```

MULX REG0
RDAX REG0,1.0000000000
SOF 1.5000000000,0.0000000000
WRAX REG1,1.0000000000 ; changed the gain here to 1.0
;----- Output
WRAX DACL,0.0000000000

```

You write the value to REGx so that it can be used elsewhere (though not shown in this example), but keep it in the accumulator so you don't have to immediately load it again for the next block's input.

Example 3: Using Sin/Cos LFO block, but not using Cos output.

As of build 837, I don't have logic to tell if a block's output is connected to another pin (although I just thought of how it could be done). So, in some cases, code is being generated in a block, even though there's nothing downstream using that result.

Before:

```

;----- LFO 0
SKP RUN ,1
WLDS 0,1,1
RDAX POTO,0.0019569472
WRAX SINO_RATE,0.0000000000
CHO RDAL,0
WRAX REG0,0.0000000000
CHO RDAL,8 ; delete this one
WRAX REG1,0.0000000000 ; and this one

```

After:

```

;----- LFO 0
SKP RUN ,1
WLDS 0,1,1
RDAX POTO,0.0019569472
WRAX SINO_RATE,0.0000000000
CHO RDAL,0
WRAX REG0,0.0000000000

```

2 instructions saved.

More examples to follow.

Top

- [Edit post \(./posting.php?mode=edit&f=32&p=1793\)](#)
- [Delete post \(./posting.php?mode=delete&f=32&p=1793\)](#)
- [Report this post \(./report.php?f=32&p=1793\)](#)
- [Warn user \(./mcp.php?i=warn&mode=warn_post&f=32&p=1793&sid=90615c931909e0633cbcd2a3573379b1\)](#)
- [Information \(./mcp.php?i=main&mode=post_details&f=32&p=1793&sid=90615c931909e0633cbcd2a3573379b1\)](#)
- [Reply with quote \(./posting.php?mode=quote&f=32&p=1793\)](#)

Re: Hand optimizing Spin ASM (#p1793)

by **Jacko** » Mon Sep 15, 2014 4:01 am

Thanks for the info, Digital Larry!

Donation sent.

Users can donate to the general upkeep of this site by following this link:

<http://holycityaudio.com/donate/> (<http://holycityaudio.com/donate/>)

Best regards, Jack

Top

- [Edit post \(./posting.php?mode=edit&f=32&p=2657\)](#)
- [Delete post \(./posting.php?mode=delete&f=32&p=2657\)](#)
- [Report this post \(./report.php?f=32&p=2657\)](#)
- [Information \(./mcp.php?i=main&mode=post_details&f=32&p=2657&sid=90615c931909e0633cbcd2a3573379b1\)](#)
- [Reply with quote \(./posting.php?mode=quote&f=32&p=2657\)](#)

Re: Hand optimizing Spin ASM (#p2657)

by **Digital Larry** » Mon Feb 01, 2016 8:07 pm

Another thing that can be done to save a few instructions is when multiple blocks have **skip run**, x initialization sections, those can all be lumped

together.

Top

- [Edit post](#) (./posting.php?mode=edit&f=32&p=2661)
- [Delete post](#) (./posting.php?mode=delete&f=32&p=2661)
- [Report this post](#) (./report.php?f=32&p=2661)
- [Information](#) (./mcp.php?i=main&mode=post_details&f=32&p=2661&sid=90615c931909e0633bcd2a3573379b1)
- [Reply with quote](#) (./posting.php?mode=quote&f=32&p=2661)

Re: Hand optimizing Spin ASM (#p2661)

by **Digital Larry** » Sun Feb 07, 2016 10:29 am

Spent some time during a walk in the sunshine yesterday thinking about how to arrange things so as to optimize "adjacent blocks" so that things like:

```
WRAX REG1, 0
; ----- next block
RDAX REG1, 1.0
```

can be in place. As it stands today, there are some cases where blocks which could be adjacent are not, due to the block sorting algorithm. So what it currently does is to make sure that later blocks are later, but not as early as possible.

Some definition of what constitutes a "block" will need to be added. For example, the output block can sometimes be thought of as two independent blocks, and so it would help to be able to pull them apart.

(brainstorming)

For each write to a register that comes at the end of a block's code, e.g.
WRAX REG15, 0

I'll want to scan down through the rest of the instructions to see where the references are to:
RDAX REG15, xx

that exist at the beginning of a block and see if that block can be "pulled up".

Now with block connection points that go to multiple destinations, there's only so much you can do, since only one of them can be adjacent to the source.

Top

Display posts from previous: Sort by

Post a reply

4 posts • Page **1** of **1**

[Return to Installing and Using SpinCAD Designer](#)

Jump to:

Quick-mod tools:

Who is online

Users browsing this forum: **Digital Larry** and 0 guests

Powered by phpBB® Forum Software © phpBB Group

Administration Control Panel