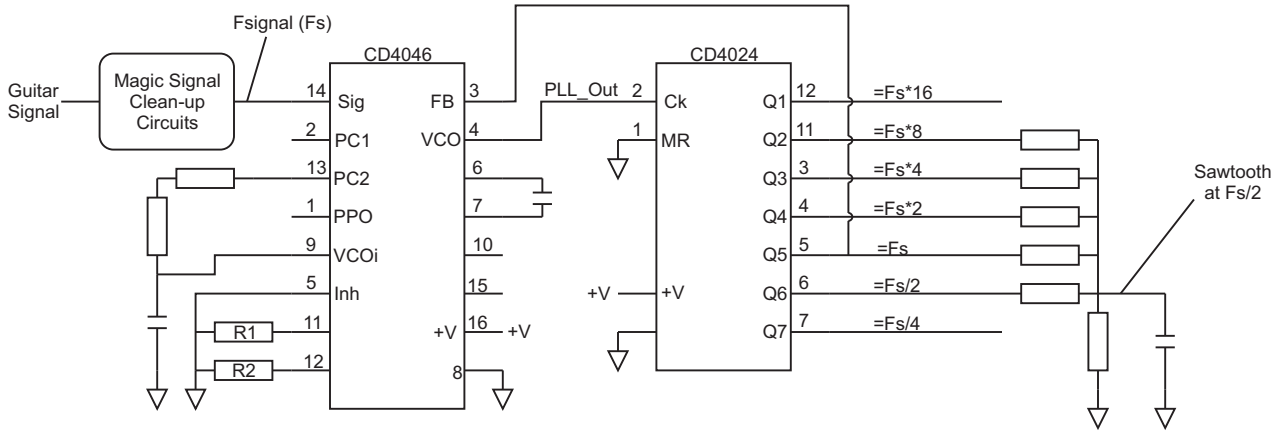
CMOS Frequency Synthesis Circuits for Guitar Pedals

You can generate musically interesting signals that follow your guitar's signal. The circuits below show some of how to do this.
Your guitar signal simply must be cleaned up into a square wave that the CMOS logic signals can see as a logic input. That means a clean transition from 0V to V+ and vice versa. If there's too much "junk" in this signal, or it doesn't go high or low enough, this won't work. That's what the Magic Signal Clean-up Circuits do. There are many ways to do this. The input circuits from Craig Anderton's Roctave Divider work well.



The fun starts with a Phase Locked Loop, a CMOS chip (CD4046) that can make its internal square wave oscillator follow a square wave input at the same frequency, and track it around as the input signal's frequency rises and falls.

We can play tricks on PLLs, by inserting a binary divider circuit (that CD4024) between the VCO and the feed back. The PLL obligingly makes its VCO the frequency that makes the divided-down signal match the input, and the divider gives us a whole slew of locked-frequency outputs.

The CD4024 by itself gives seven octave-related outputs. So if the PLL makes the output at Q5 be the same frequency as the input frequency at the PLL, you get two octaves down and four octaves up from the input signal.

But they're still square waves and sound kind of harsh. A little signal synthesis helps. Those resistors after the CD4024 sum up five of the divider outputs. If you juggle the resistor values right, you get a stair-stepped approximation of a sawtooth wave. That happens to be an only slightly-crude approximation to the sound that strings make anyway. Not perfect, but much closer than a square wave.

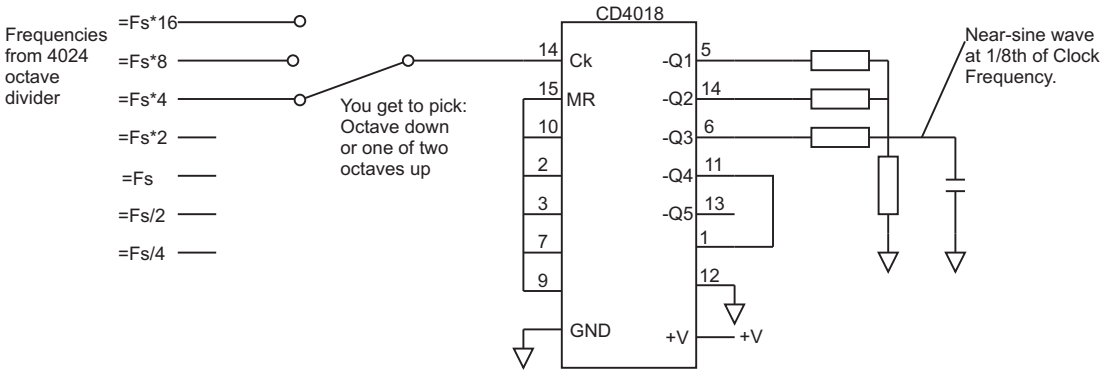
So right there at the junction of those resistors you get a string-ish sounding approximation to one octave down from your guitar signal in. The resistor and capacitor to ground help smooth off some of the rough edges in the stairsteps. This signal is likely to be a little hot for guitar level pedals or amps that come after this circuit, but you can freely reduce this resistor to cut the signal level a bit.

As they say - or used to say - on late-night TV commercials, wait! There's more.

If you like an even less buzzy octave up or down, you can do another kind of synthesis to get a nearly-pure sine wave. That's what the circuit below does. The CD4018 is a shift register, and it's set up to make a semi-complicated set of digital patterns that it recirculates through itself over and over continuously. The resistor adder network is used again to sum these digital waveforms up, but they have different, odd values from the binary divider summers. In this case, the resistors and the funny internal waveform make a digital approximation to a sine wave. In this case, the output is a near-sine wave at 1/8th of the input clock frequency.

There is some math behind this, but it does work.

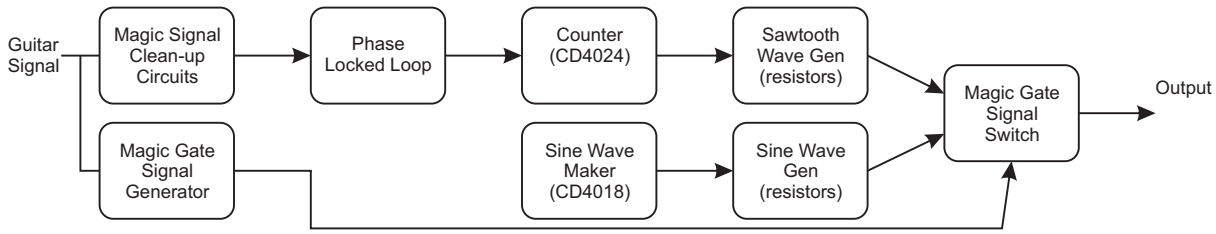
We can use the many outputs of the PLL's binary divider again to provide the clock to the sine maker. Conveniently, we have four, eight and sixteen times the input signal available, so this little add-on can make nearly-sine waves at one octave down and one or two octaves up from the original guitar signal.



There are more issues to be cleaned up before this is a functioning pedal. The PLL runs all the time, so if you don't mute it somehow, it will always be putting out maximum signal at whatever noise signal it locks onto or the min/max of its VCO range. Unless you're a strong fan of random sounding noise-music, that will tiresome, very quickly. The first step needed to make this useful is a gate.

A gate does the simple function of turning on when there is real guitar signal coming in, and off when there is not. Don't get the impression that designing a good gate for guitar signals is easy - it's not. But even a modest one may be all you need. For now, I'll just represent the gate function as a magic box, just like I (frustratingly, I know) did for the input signal clean up circuits.

The gate signal is just a logic signal that tells the gate switch when to turn signal on and pass it to the output, or when to keep signal from going through.

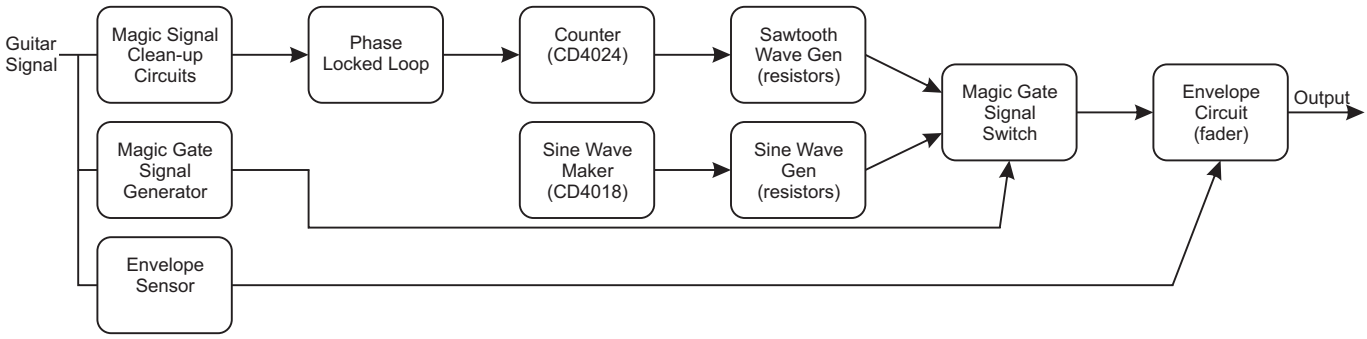


About now, you're probably thinking "wait - this is getting complicated." You're right, it is. This is why there aren't a plethora of good octave and harmony generating pedals on the market or in DIY forums and download-me-now layouts and such. Doing a simple if ugly version is easy. Making it a really usable musical tool takes some work.

The gate cleans things up so that you only have generated sound coming out when your guitar is making sound. But it's completely at one output level, like an organ. It would be more useful if it had some dynamics, some attack and fade out to the sounds it makes, like a guitar does.

That needs an envelope generator.

There are several types of useful envelope generators, but the most obviously useful one is the kind that simply senses how loud your guitar signal is, and makes the output of the generated octaves be the same or just a hair less. It would look something like this:



The envelope sensor makes a signal that is proportional to your guitar's loudness, and then sends it to the envelope circuit, which is an electronic volume control or fader. When the envelope is high, the volume control is high, when the envelope is low, the output volume is low. You may be wondering why if we have an envelope follower we need a gate. It's because gates are on/off, not messing about almost on or off. You want the space between notes to be completely silent.

There are some envelope circuits that can accept the gate signal directly and turn full off, so there may be some economies there.

As a final enhancement, it is possible to make the CD4018 divide not by binary values, but by whole numbers, like 3, 5, 6, 7, etc. The E&MM Harmony Generator used this trick to make a circuit that output thirds, fourths, fifths and such in harmony with the input signal.

See - it only takes a little more circuitry...