SCHLAPPI ENGINEERING

NIBBLER MANUAL

Technical Information

Voltage Levels

The Nibbler is designed for compatibility with Eurorack voltage standards. Inputs and outputs are voltage and current protected and should not but damaged by any level within the Eurorack ecosystem (-12V to +12v, or 24v peak to peak).

SIGNAL TYPE	LEVEL	Notes
Gate outputs	0 or 10V	
Stepped outputs	0 to 10V	
Gate inputs	2.8V threshold	Comparator input stage triggers around 2,8V

Current Draw

+12V: 37mA

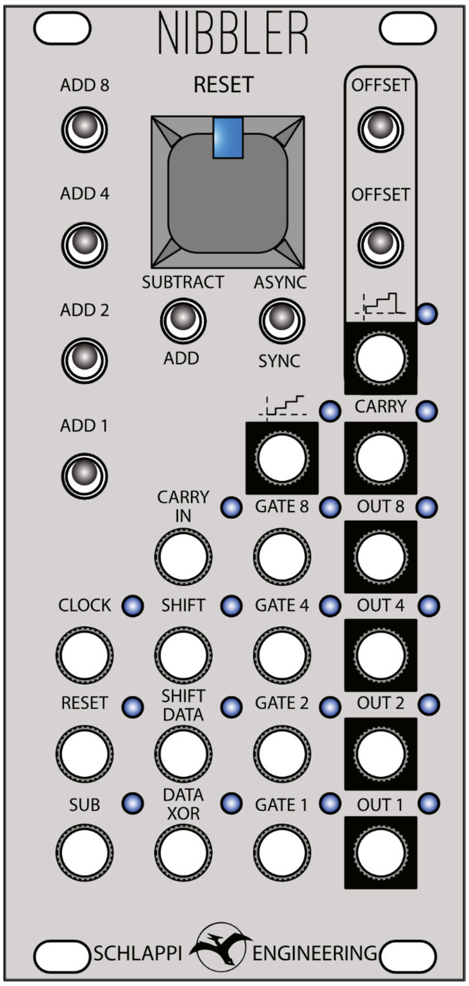
-12V: 17mA

Module Description and Features

Introduction

The Nibbler is a four bit digital accumulator based on CMOS logic. What this means is that it counts in binary from zero to fifteen, with inputs and outputs for individual bits as well as stepped voltage outputs (digital to analog converters). It does this with individual logic chips instead of a CPU.

The concept here is that counting in binary (and its expression in bits) is inherently musical, and we can use it to create both rhythms and modulation voltages (or melodies).



The interface is designed to to be playable, with switches for the binary counting as well as two mode switches and two phase offset switches (so the second stepped analog output can be offset in phase from the first). There is also a large Cherry MX Brown keyboard switch for resetting the register to zero.

Controls

Switches

CONTROL	DESCRIPTION
ADD 8	Adds 8 to the register
ADD 4	Adds 4 to the register
ADD 2	Adds 2 to the register
ADD 1	Adds 1 to the register
SUBTRACT/ ADD	Determines if the number formed by the ADD SWITCHES is added or subtracted from the number already in the register
ASYNC/ SYNC	Determines if the output is updated only on a rising clock pulse (SYNC) or every time an input is received (ASYNC)
OFFSET	These two switches together set a phase offset for the second stepped voltage. It can be 0º (both down), 45º (only bottom switch up), 90º (only top switch up), or 180º (both switches up)
OFFSET	

Button

CONTROL	DESCRIPTION
RESET	Clears the register, setting all outputs to 0

Inputs

All inputs are logic inputs with a threshold of around 2.8V that trigger on the rising edge.

INPUT	Description
CLOCK	Clock input, also used for audio rate frequency division purposes. Necessary for most operations.
RESET	Clears the register, setting all outputs to 0, AC coupled so it can be used as a hard sync input at audio rate
SUB	This input interacts with the SUBTRACT ADD switch to change the state to the opposite of the current setting.

All the other inputs are affected by the ASYNC/SYNC control to either only have an affect on the rising edge of the clock or immediately.

These inputs control the rate of counting.

INPUT	Description
CARRY IN	Intended for chaining multiple Nibblers, to make a larger register by patching a CARRY OUT to a carry in. Effectively the same as GATE 1.
GATE 1	Adds with the ADD 1 switch to set the rate of counting.
GATE 2	Adds with the ADD 2 switch to set the rate of counting.
GATE 4	Adds with the ADD 4 switch to set the rate of counting.
GATE 8	Adds with the ADD 5 switch to set the rate of counting.

These are the shift register inputs. The ASYNC/SYNC control similarly affects whether it will shift only on a clock or any time a pulse is received.

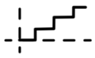
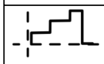
INPUT	Description
SHIFT	While SHIFT is high in SYNC mode any clock pulse will shift the contents of the register up one, or if in ASYNC mode it will shift on any pulse on this input.
SHIFT DATA	Replaces the input of the shift register. With no input the top bit (OUT *)cycles around and enters from bottom.
DATA XOR	Performs an XOR function with whatever data is at the input to the shift register

Outputs

Gate outputs are either 0 or approximately 10V

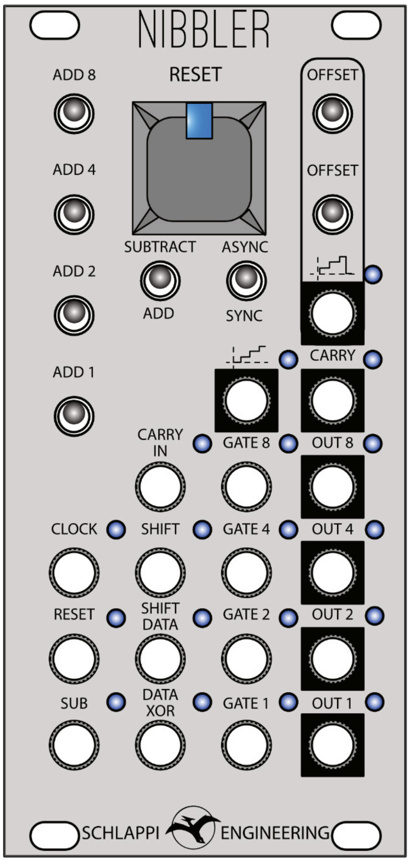
NAME	DESCRIPTION
CARRY	The CARRY out goes high for one clock pulse when the register overflows. You can use this to chain additional Nibblers for a bigger register, or as another gate output. This is probably the output to use if you are using Nibbler as a clock divider.
OUT 8	Gate output for the top bit of the register, which represents 8
OUT 4	Gate out put for the register bit that represents 4
OUT 2	Gate output for the register bit that represents 2
OUT 1	Gate output for the bottom bit of the register, which represents 1

Analog stepped voltages output 0 to 10V

LABEL	NAME	DESCRIPTION
	STEPPED OUT 1	A weighted sum of the register bits as a stepped analog voltage
	STEPPED OUT 2	The register bits summed with the two offset switches to create a static phase offset

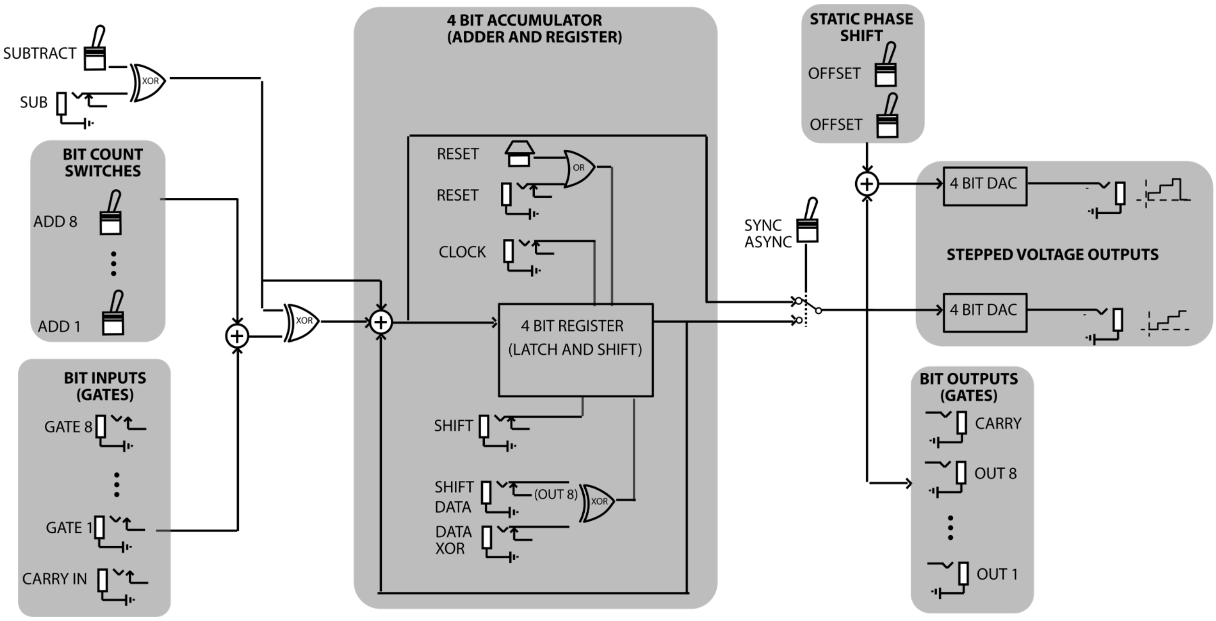
Indicators

All inputs and outputs have blue LEDs indicating their current state, at audio rate they may show a solid blue. The reset button also has an LED underneath it.



How It Works

Block Diagram

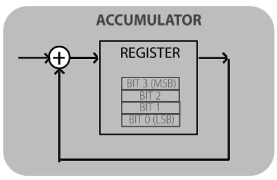


Binary and Accumulators

Binary is a number representation which only uses ones and zeros, and each position corresponds to a power of two. With a four bit word we have 1, 2, 4, and 8, which together can add up to 15.

ADD 8	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
ADD 4	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
ADD 2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
ADD 1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

The combination of switches and gate are the input for a 4 bit binary word. This will be a number between 0 and 15. This is added to the number already present in the register (a 4 bit memory unit) and then stored back in the register. This configuration of an adder and a register is known as an accumulator.

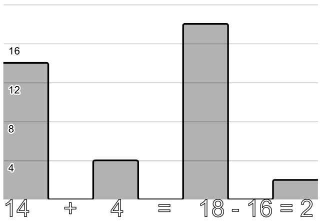


Accumulators are a fundamental digital building block and have many interesting properties, one of which is that it is the digital equivalent of an integrator (the mathematical operation) and is a big part of most filter or oscillator designs.

This particular register also has a bit shifting operation built in, which rotates the bits from lowest to highest at each clock pulse while shift is high (in synchronous mode).

OUT 8	0		0		0		1		0
OUT 4	0		0		1		0		0
OUT 2	0	↑	1	↑	0	↑	0	↑	0
OUT 1	1		0		0		0		1

If the number in the register goes above 15 it overflows and wraps around. For example if there is 14 in the register and you add 4 then the new value will be 2. You can think of this as a modulo 16 operation.



This has two cool properties: One is that if you are counting by one, then each higher bit will be half the frequency as the previous one, allowing for use as a clock divider or octave down effect. The other is that if you increment the counter (accumulator) by an odd number, then you will get a waveform (or sequence) that will keep wrapping around with an offset and take some time to repeat. See the waveforms page.

Synchronous/Asynchronous Operation

The “SYNC / ASYNC” switch determines if the output is taken from the register, which will update only on a rising clock pulse, or from the adder which will change immediately. This can be useful for wonky patterns or changing the audio rate effect.

For clean sounding rhythms and modulation it probably makes sense to keep this switch set to SYNC, however at audio rate you can think about modulating the gate inputs as frequency modulation. Frequency modulation inherently filters modulation close to or above the carrier (in this case the clock signal).

In the ASYNC mode (taking the output from the asynchronous adder) then you will be getting a combined phase and frequency modulation effect at lower modulation frequencies and at higher frequencies the phase modulation will dominate. At audio rate which one you want will probably be determined by whether you want that filter effect or not, try both!

See the Three Body manual for more about phase and frequency modulation. The same principles all apply here, except the clock input of the Three Body’s oscillators is a fixed 12.5MHz and the accumulator is 36 bits deep. This allows for many orders of magnitude higher precision and fidelity, however the point here is to interact with the bits directly.

The ASYNC mode also has a somewhat unusual normalization, the SHIFT and CLOCK inputs are XOR’ed together and sent to the clock of the shift register.

This means that in ASYNC mode you do not need to use the CLOCK input to use the SHIFT input, making the shift register functionality independent of the accumulator functionality. This can then be used for a Rungler patch as detailed in the patch section of the manual.

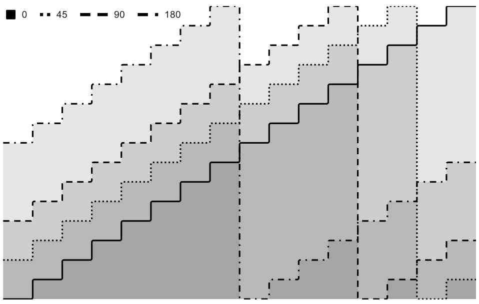
The SUBTRACT switch (and related gate input) alters the direction of counting. If you look at the waveforms page you will see that that can also be achieved by changing the frequency word (setting of the switches). In this case they are primarily intended as performance controls to allow for more dynamic sequences.

Phase Offset Switches

The two offset switches can be used to create a second stepped voltage offset in phase. The four offsets are shown below. At audio rate this won't make much of a difference (unless you are doing oscillographics) but at LFO rate for modulation this can be used so one sequence is at it's highest value when the other is at its lowest (180 degrees) or offset just slightly (45 degrees) or 90 degrees off.

The intention here was to have two modulation voltages for use with the Three Body and other stereo processes. It would also make a lot of sense to feed into a multichannel quantizer and get and effect of multiple melody lines following each

Lower offset switch	Upper offset switch	Degree offset	Numerical offset
down	down	0	0
up	down	45	2
down	up	90	4
up	up	180	8

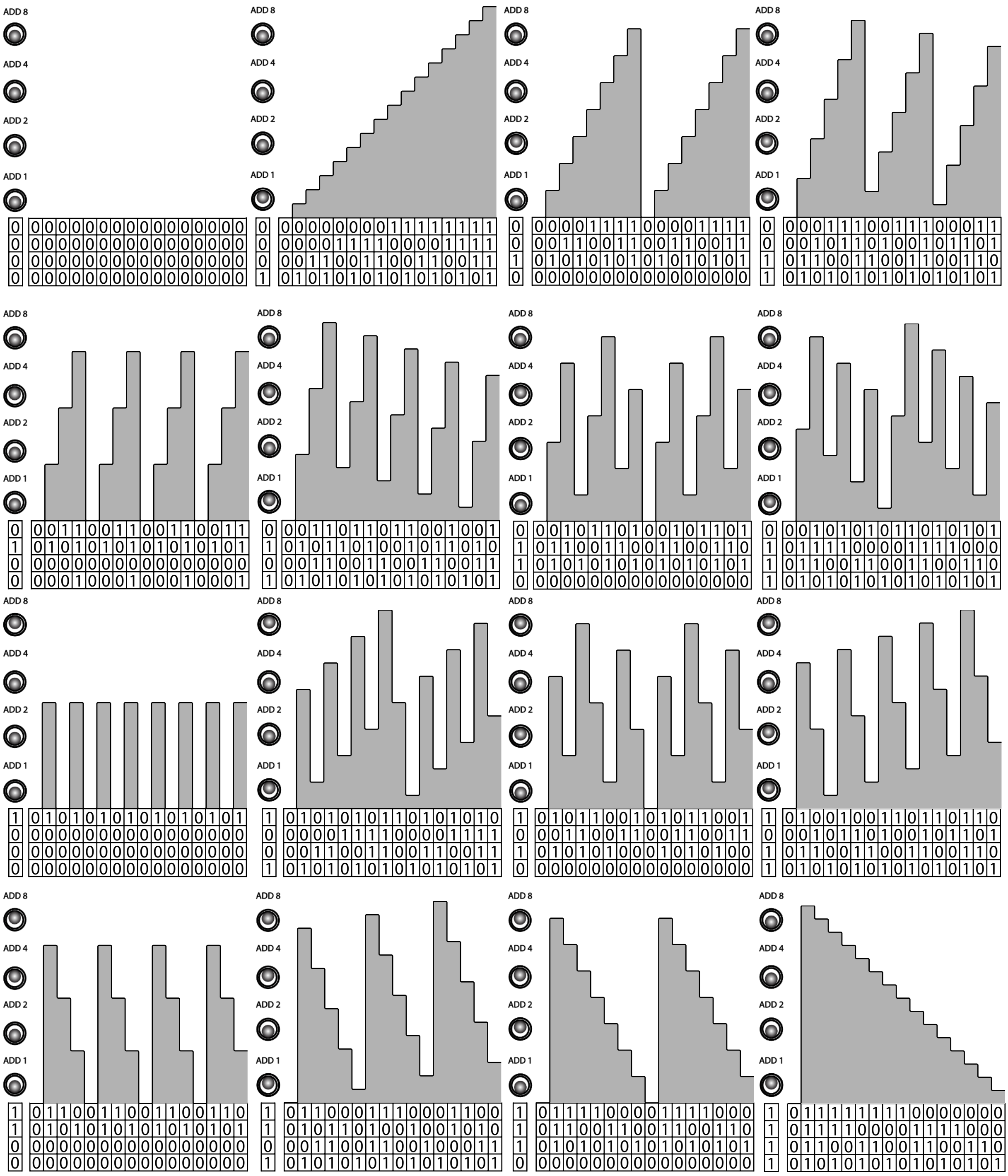


Longer sequences can be created by modulating the gate inputs.

Waveforms

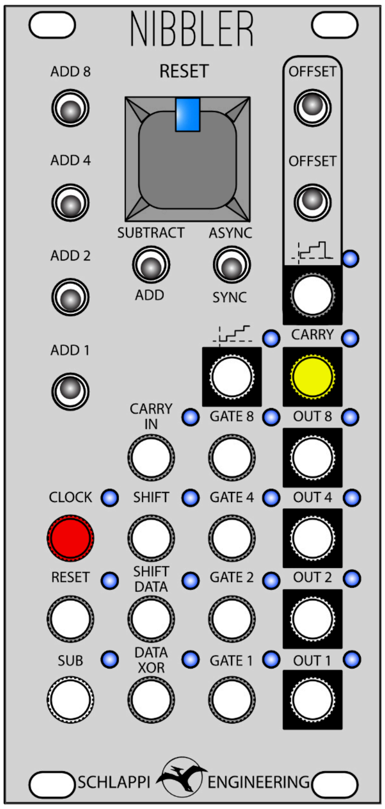
The four switches of the nibbler can be thought of as defining 16 states, waveforms, or sequences (depending on how you are using it). They are shown below (counting up, from left to right) along with the binary representation (which would also mirror the gate outputs).

You can see that counting by zero yields nothing, while counting by one creates a rising ramp, counting by fifteen creates a falling ramp, and the waveforms in between are mirrored around counting by 8, which creates a series of pulses at half of full-scale amplitude and half the rate of the incoming clock.



Patches to start exploring with:

Clock Divider



Set all switches down except for ADD 1

This means it will be in synchronous adder mode and incrementing by 1 with each clock pulse

Patch a clock signal into the CLOCK input

Use the CARRY as your clock divider output

This should give a clock division of 16

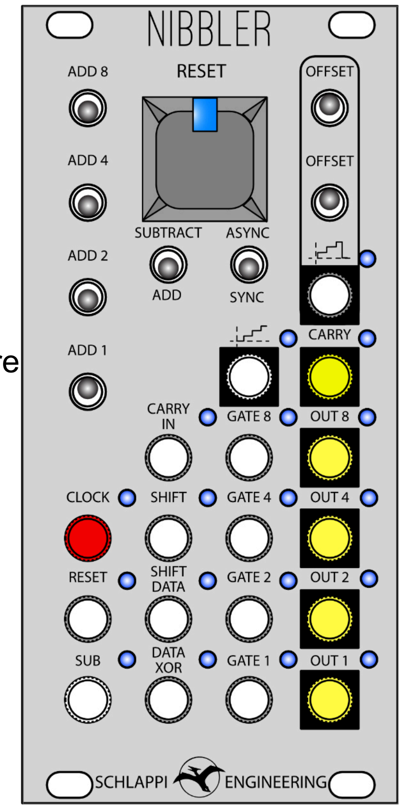
If you use the ADD 2 switch instead of ADD 1 you will get a clock division of 8

ADD 4 by itself will give you a division of 4

ADD 8 will give you a division of 2

If you combine switches into odd values you will get a division of that number over sixteen. For example ADD 1 and ADD 2 together will give you a division of 3/16.

Clock Divider Drum Sequencer



Set the controls the same as the clock divider patch but instead of just using the CARRY out use all of the gate outputs.

Start by patching the OUT 8 to a kick drum sound

Use OUT 4 to trigger a snare or tom

Use OUT 1 to trigger a high hat

This will have the kick sound on every 16th clock, the snare on every 8th, and the hi hat on every other clock.

KICK									1							
SNARE					1								1			
			1				1				1				1	
HAT		1		1		1		1		1		1		1		1

After this, explore what rhythms different combinations of the switches create.

Things will become much more interesting if another Nibbler or clock divider is used to modulate the gate inputs.

Also try using the stepped outs as a clocked modulation signal within the patch.

Triangle Wave

Set the Nibbler to count by 1 by flipping the ADD 1 switch up and all other switches down

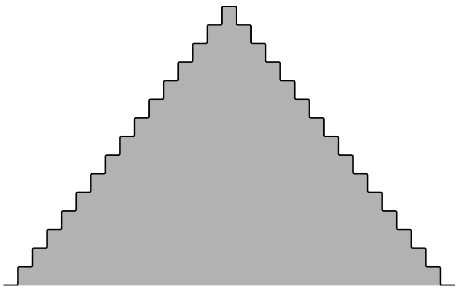
Patch a clock into CLOCK

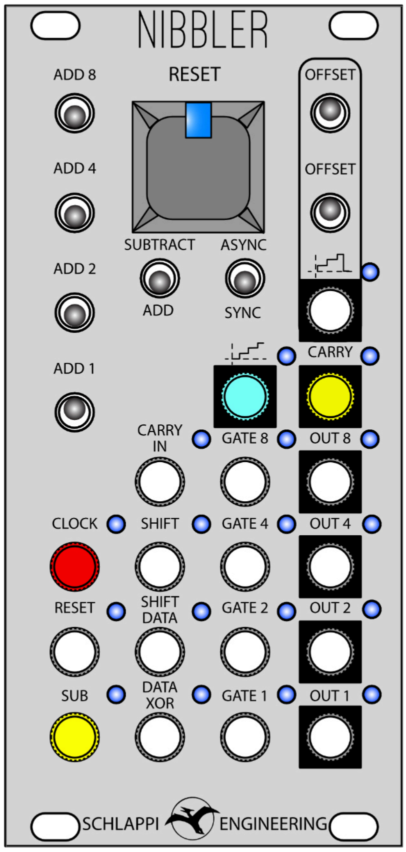
Use STEPPED OUTPUT 1 as your output

Patch CARRY out to SUB in

When the CARRY output goes high it will trigger the subtract input, changing the direction of counting

This will create a triangle shape at 1/32 the frequency of the input clock





Frequency Divider (or Subharmonic Generator)

Patch an audio rate signal into the clock input

Listen to the first STEPPED OUT

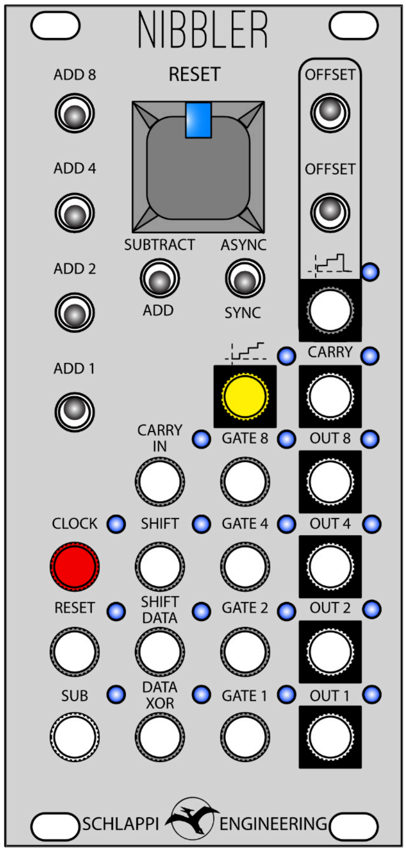
If only ADD 1 is up you will get a ramp wave at 1/16 of the input frequency.

See the waveforms page for the waveforms that will be created by other switch combinations (or see above in creating a triangle wave)

Send signals into the gate inputs to create timbre and pitch changes (since it will change both the frequency and waveform). This is a good wave to create chiptune arpeggios and basslines.

Using the gate outputs as well will give you harmonically related pulse waves.

The CARRY out will give you a very thin pulse wave 1 clock pulse long.



Shift Register Noise

Patch an audio rate signal into the CLOCK input

Patch another audio rate signal into the SHIFT input

Set the ADD switches to some combination that include both down positions and up positions

Listen to a stepped output

Try this in both SYNC and ASYNC mode

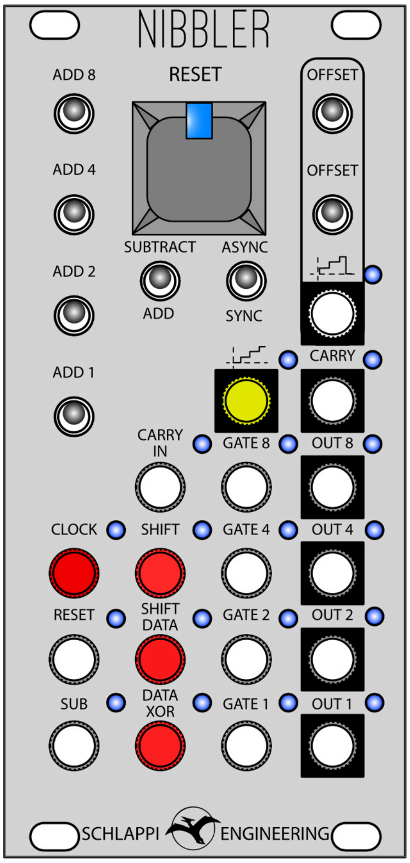
In SYNC mode you will get a sample rate reduction and filtering effect with by slowing down the clock input.

In ASYNC mode you will get another flavor of noise with a bit more high end to it

By sending a third signal into the DATA XOR input you will get another ring modulation type effect

You can also replace the data in the register bu inserting a signal into the SHIFT input

The shift registers can be chained with multiple Nibblers by patching the top bit (OUT 8) into the next Nibbler's SHIFT DATA input. This can be done in a circular chain.



Rungler Variant

All switches down except ASYNC should be up

Patch OUT 8 into DATA XOR

Patch one oscillator into SHIFT

Patch another oscillator into SHIFT DATA

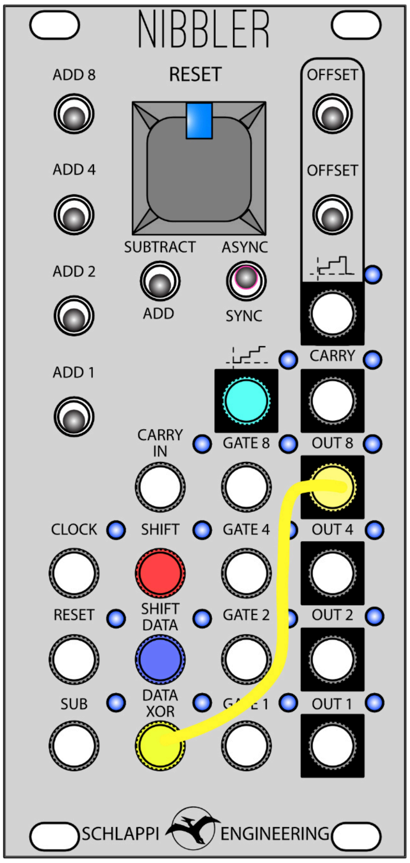
Use the stepped voltage outputs

Depending on the rate of the oscillators the output can be either LFO or audio rate and it can be patched into back into the oscillator frequency modulation inputs for a Benjolin like patch.

The Nibbler is only 4 bits instead of 8 so the patterns will be a bit different than Rob Hordijk's Rungler but it is an interesting way to generate some psuedo random patterns.

The ADD switches will also change the pattern as will all inputs.

See next page for more.



Simple Benjolin Variant

All switches down except ASYNC should be up

Patch OUT 8 into DATA XOR

Patch the right oscillator into SHIFT

Patch the left oscillator into SHIFT DATA

Patch a stepped voltage output into the FM input on both oscillators.

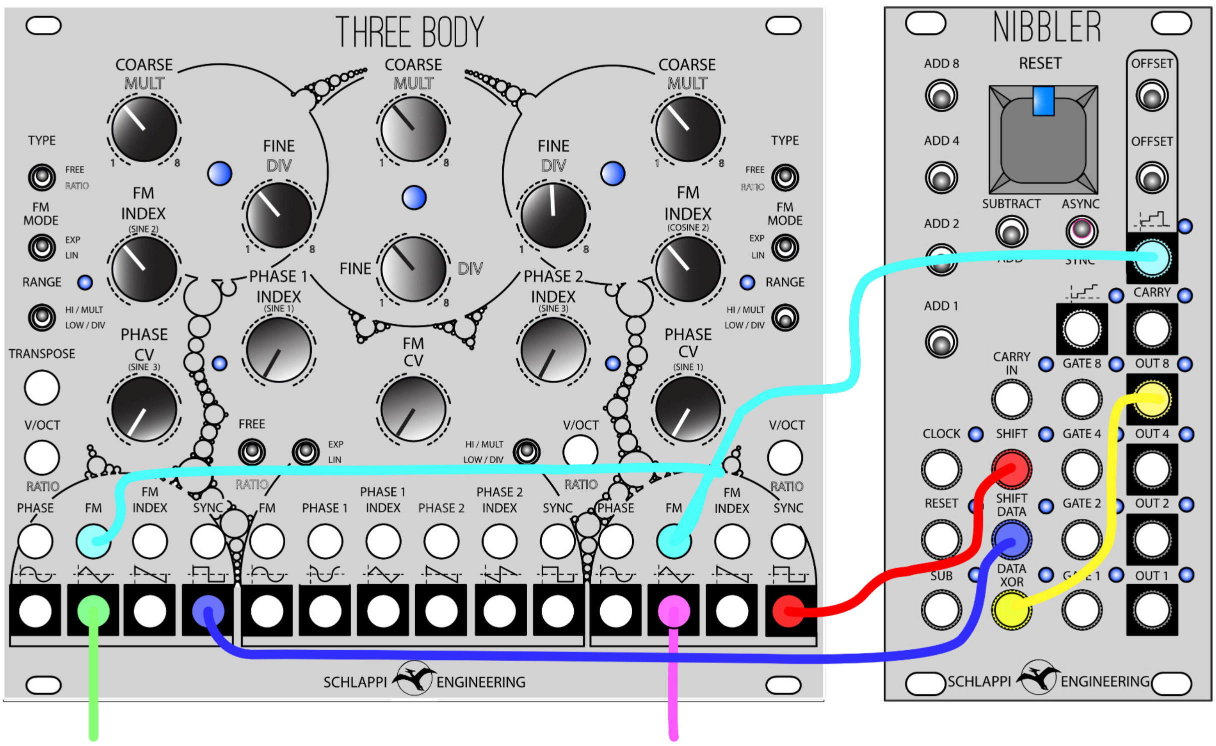
Patch a waveform outputs from the oscillators as well as (optionally) the stepped voltage output from the nibbler into an audio mixer.

The true Benjolin would run these waveforms into a filter which would also be modulated by the stepped voltage and an oscillator.

Experiment with the amount of FM and rate of the oscillators for different melodies and rhythms.

The bit switches on the nibbler will all alter the pattern as will the offset switches if that stepped output is being used.

To make things even wilder use the center output on the three body to modulate any of the inputs on the Nibbler, I had particularly good luck with the CLOCK or GATE 1 inputs at LFO rates.



Sequencer (or Arpeggiator)

Patch a clock into the CLOCK input

Patch the STEPPED OUT 1 into a frequency input of an oscillator (or through an attenuator into the RATIO input on the Three Body)

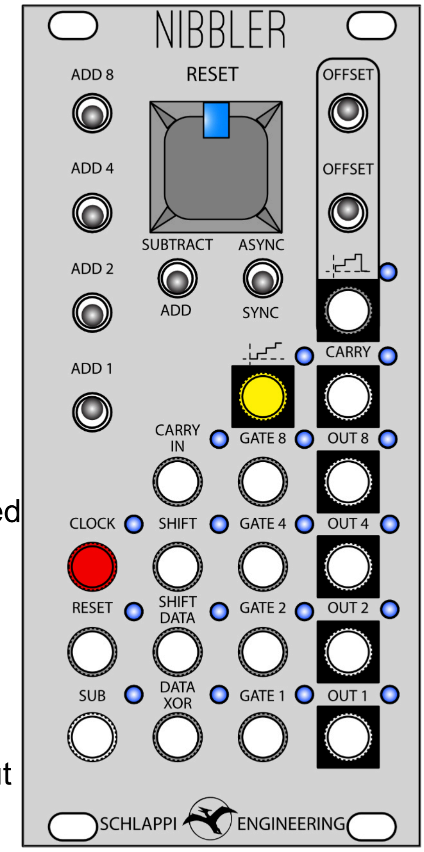
Each combination of the ADD switches will create a different sequence (see the waveforms page)

These sequences can become dynamic or ever-changing by patching another sequencer, clock divider, or other signal into the GATE or SHIFT inputs.

Two nibblers with the OUTS and GATE ins cross patched (OUT 8 to GATE 2 and OUT 2 to GATE 8 for example) creates some nice generative type sequences

Simultaneously using the gate outs (like the Clock Divider Drum Sequencer patch) will turn the Nibbler into a full sequencer.

If being used as an arpeggiator, try using the gate output of a keyboard into the RESET input to restart the arpeggio with each new note played

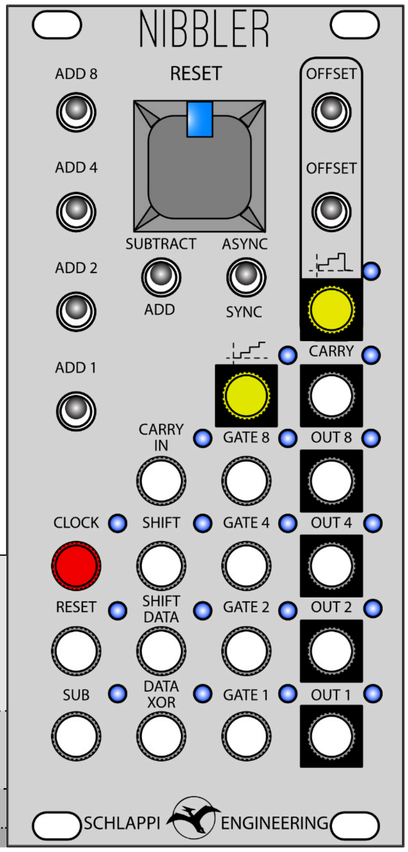


Phase Offset Sequences

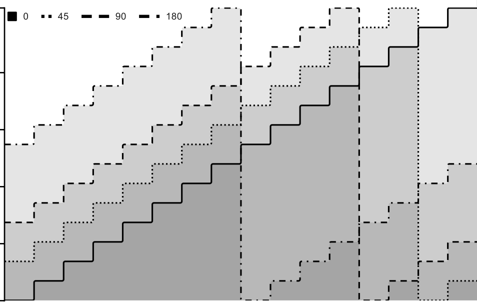
This is the same set of patches as the above Sequencer or Arpeggiator except now we use both stepped outputs.

To get a feeling for what this can do run it at LFO rates and patch one stepped output into one oscillator (or Three Body ratio) and then send the other into the frequency input of another oscillator.

With both switches down you will get the same sequence on both, flipping up just the bottom switch will get a slight offset (45 degrees), just the top will get you 90 degrees, and both at once will put them 180 degrees apart (so that when one is at it's lowest point the other is at it's highest.



Lower offset switch	Upper offset switch	Degree offset
down	down	0
up	down	45
down	up	90
up	up	180



Cross Patched Nibblers

Send the same CLOCK signal to two Nibblers

Set them both to SYNC mode

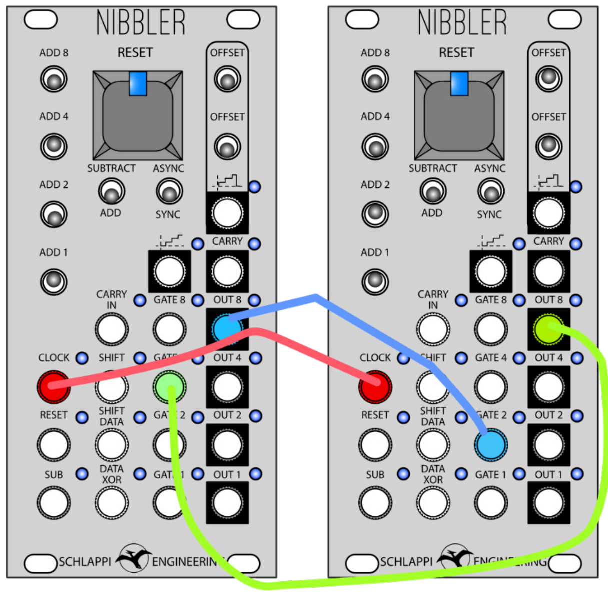
Try cross patching the gate outputs and inputs

Try hitting the reset on only one of them

You will find that this creates a different sequence each time you do it

Use this patch along with any of the other patches described in this manual, you will find it is particularly good for create more complex rhythms or melodies that may not repeat for quite some time or change somewhat unexpectedly

If you only have one Nibbler try using any logic you have to modulate the gate inputs for a dynamic sequence



Chaining Nibblers

With two Nibblers patch the same clock signal to each

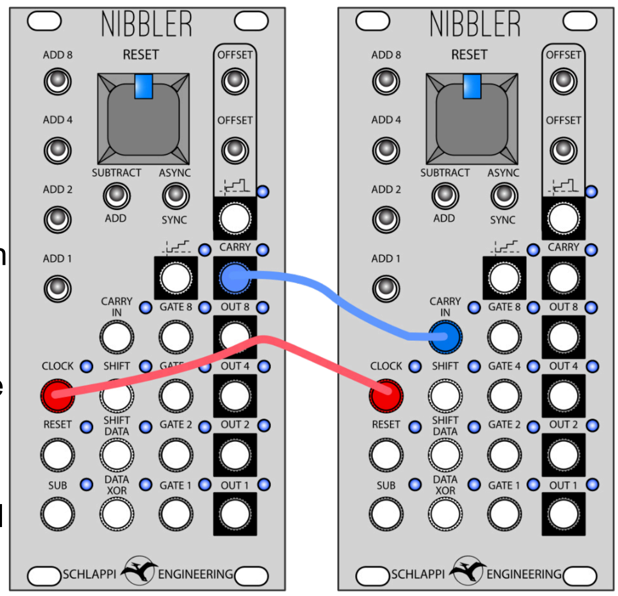
Patch the CARRY out of the first one to the CARRY IN of the second.

Now you have an 8 bit deep register which can divide up to 256

The first Nibbler will control the lower 4 bits and the second Nibbler will control the upper 4 bits.

If you wanted to divide by 256 you would set the first Nibbler to add by 1, and put all switches on the second down (so it is adding by 0).

Each stepped out will still only be 4 bits, you could mix them together so that the first nibblers output is 1/32 of the second nibblers to achieve 8 bits, but it might be a fiddly thing to try to accomplish.



Contact Info:

If you have any questions please contact Eric Schlappi at: eric@schlappiengineering.
com