Synthesis on the Nord Stage 2

Version 1.1 DRAFT

Contents

Introduction

This is a guide to using the synth section on the Clavia **Nord Stage 2** and **Nord Stage 2EX** keyboard instruments. For the rest of this guide I'll refer to both as the **NS2**, as both instruments have identical features (the EX simply has more piano memory, which is not relevant to this guide).

It is not an official document, and is not affiliated with or endorsed by Clavia. There may be several mistakes – please feel free to let me know via an instant message on the forum above.
You use this guide at your own risk. If you die of boredom or blow your mind, I am in no way held responsible.

Synth manuals are usually worded in such a way that even the writer sounds bored. I'm going to use the same (standard) terminology, but I'll try to make it easier to digest.

I'm assuming you have read the ordNS2 user manual, and you are familiar with the basic operation like activating/deactivating different sound sections (Organ, Piano, Synth, External), creating keyboard zones/splits, how to switch between slots A & B or activate them both at once, and how to store programs.

I've tried to write this so it makes sense to someone who's never owned a synth before, but still useful to people who know quite a bit already. Also, the FM section is aimed equally at owners who are new to FM and also those who have previous experience with Yamaha DX synths.

Conventions used in this guide

General synth terms like **LFO** are shown in bold.

✍ *Hints, tips and other points of interest look like this*

☞ When you are expected to do something on the NS2, you'll see the red hand symbol at the start of the line.

Functions or features specific to the NS2, like **MOD ENV** are shown in bold red.

Techy bit about modulation and stuff

There are a few concepts used in synthesis that are really important to understand, or you won't get anything later – especially when we get into FM synthesis, so lets get this out of the way now.

An **OSCILLATOR** (OSC) is something that changes from what state to another in a repetitive way. In a synthesizer, an oscillator is a device that varies the voltage it outputs – imagine turning a dimmer switch backwards and forwards really quickly, and you get the idea.
When a wave on the sea moves up and down, it oscillates slowly. If you hit a tuning fork, the prongs are oscillating very quickly. The faster they oscillate, the higher the pitch.

If you attached a tiny pen to the tip of the tuning fork, and drew it along a sheet of paper, it would make a wiggly line in the shape of a perfect sine wave.
If you mark the piece of paper when you start moving the tuning fork, and again after 1 second, the distance between the two marks shows the number of waves in 1 second. A low-pitched tuning fork would have waves appearing less frequently than a high-pitched one.
How frequent the waves are is called the **FREQUENCY.**

✍ *This experiment was actually done nearly 200 years ago by a German guy called Helmholtz.*

A tuning fork is tuned to note A, which has a frequency of 440Hz – this means there are 440 waves in 1 second. Waves per second is called Hertz (Hz).

Hit a key on a piano, the sound goes quickly from zero volume to a certain level (depending how hard you hit it), then slowly falls back to zero. This amount of volume level at any point is called **AMPLITUDE**.

Most synths have a modulation wheel (or similar control such as a joystick). A popular misconception is that it adds vibrato to a sound. Not necessarily. It's just a control which you might use for making vibrato… you will see it can do lots of things, especially on a NS2.

MODULATION just means changing or tweaking.

Hold a key on an organ. If you turn the volume control slowly backwards and forwards, the sound changes in volume up and down over time. It's modulating its amplitude so we call it **AMPLITUDE MODULATION**.

If a sound changes its pitch over time instead of its volume, it's modulating its frequency. We call it **FREQUENCY MODULATION** or **FM** for short.

If these changes in frequency happen in a repetitive way like a police siren, then the change in pitch between low to high and back again is quite slow. It’s being modulated by a low frequency oscillator or **LFO**.
If the frequency of this modulating oscillator starts to get faster, the ear can’t detect the differences and you hear a new tone, the LFO has become audible just like a normal oscillator and so we call it self-oscillating.
What if we don’t want the amplitude or frequency to wobble up and down?
What we want to make a percussive sound like a piano is for the amplitude to go up to maximum level really quickly, then slowly drop back to zero. If you could communicate with the oscillator, you could write those instructions on a piece of paper:

“Start at zero, go up to maximum volume in 1/1000th of a second, then drop back to zero over 5 seconds.”

You could put this piece of paper in an envelope and give it to the oscillator.
That’s what an **ENVELOPE** is. A set of instructions:

ATTACK:	how fast to go from zero to maximum
DECAY:	how fast to go from maximum to zero while they key is held down
RELEASE:	how fast to go from whatever level you are at to zero when the key is released. (If the decay has already got to zero, the release will have no effect.)

✍ *Other synthesisers often have a SUSTAIN stage between decay and release, some have a DELAY stage before attack, and some have multiple stages. Since the NS2 only has these 3 stages, I won’t go into what the others do. We can live without them.*

Because the synth generates the envelope, it’s called an **ENVELOPE GENERATOR** or **EG** for short.

The other important thing to say about envelopes, is that you can use them for different things. They are instructions for modulating (changing) things, so we call them **MODULATION ENVELOPES** or **MOD ENV** for short.

Envelopes can be used to modulate amplitude (AEG), pitch (PEG), filter (FEG) or even other modulators. We’ll come back to some of these later…

Overview of Synth Section

If you are new to the NS2 or unfamiliar with the Synth Section, jump straight to Chapter 3 of the NS2 User Manual, page 12. It covers some essential features that you need to know before going any further.

Common Features

Volume control, keyboard zones, octave shift etc. Read all about these in Chapter 5 of the NS2 User Manual.

The Synth controls are covered in Chapter 8 (page 29) of the NS2 user manual. I'll briefly summarise them here to save you flipping between documents, but there is no point repeating the manual. Seriously - RTFM.

Arpeggiator, Voice Mode, Unison, etc.

You read Chapter 3 of the user manual didn't you? You need to know how each of these works. One of the most important features when making your own synth sounds is **Unison**.

OSC Section

OSC is short for OSCILLATOR. See page 2.
This is the starting point for any new sound, right after you have done a "**Sound Init**".

Waveform Selector Button

You have to start with something that makes a noise. It could be an electronically generated waveform (SQUARE, SAW or TRIANGLE icons), a sampled sound (SAMP) that you loaded in, frequency modulated waveforms (FM) or wavetable waveforms (WAVE).

Waveform Selector Dial

I had to dig around the manual to find out what they call this, as it is not labeled (it's the knob under the red display), but it allows various options for each waveform type – cycling through the sampled sounds being the most obvious use.

SHAPE Knob

This knob is labeled a little misleadingly, as it does different things depending on the type of waveform you are using. Because of that, I'm not going to describe it here, but come back to it later.

SHAPE MOD Knob

When you turn the shape knob with your hand, you are ***modulating*** the shape of the wave.
Instead of doing this modulation with your hand on the shape knob, you might want to automate this tweak so it happens whenever you play a key. That's what this knob does. We'll come back to how that all happens later.

AMP ENV Section

AMP is short for AMPlitude and ENV is short for envelope. I covered this on page 3 of this very document. You didn't read it? You skipped right to the first page with pictures didn't you? AMP ENV is dedicated to controlling the amplitude (volume) of the sound.

MOD ENV Section

Also on page 3 of this document. MOD ENV can control different types of modulation, such as the amount of filter to apply etc.

Filter Section

This is where it all happens when you are making sounds using subtractive synthesis, where you layer up several OSC waveforms, and then FILTER away the harmonics you don't want. Page 32 of the NS2 User Manual has lots of nice diagrams to look at. We'll cover all of this in some depth in the Subtractive Synthesis chapter, so for now, just be aware that this whole section is called the filter.

The surprising news is that in FM synthesis you won't be using it much, if it all.

Subtractive/Analog Synthesis

If you have never really 'got' synthesis, this is the best place to start. The process of making sounds is quite intuitive.

Menu of sounds we will create

- Some basic acoustic instrument – type sounds: **Strings**, **Reed**, **Flute**, **Brass, Pluck**. Don't expect these to be awesome programs in their own right, but they will show you some important envelope characteristics.
- **SuperSaw** – classic analog staple sound
- **Disco Pop** – demonstrates the use of filter resonance.
- **Pad** – we'll finish up with a really fat and rich pad sound.

Coming soon... for now, skip ahead to FM Synthesis

FM Synthesis

If you are the impatient sort, here's a quick heads-up: we are going to make a bunch of representative sounds (patches/programs/etc.) that should give you enough of an understanding of how the FM synth works to start producing good sounds of your own.
I'm not going to go into every possible algorithm and I'm not going to get any more 'techy' than is necessary.
If you want an in-depth book on the science of sound, there are plenty. Also, it's worth me caveating that my knowledge on this topic is about just enough to make the sounds I want: there are some infinitely smarter folk on the *Unofficial Nord User Forum*, but they are probably busier than me.

Here's the menu of FM sounds we'll create:

- **Hammond Organ** type sound – it's a great starting point as you have a the organ section Hammond to compare it with. You may even have a real one. Lucky you.
- **E Piano** – slightly harder than the Hammond, but again, great for comparison as you have the Piano section right there.
- **FM Bass** – this is one of the things FM is really good for.
- **Metallic/bell sound** – this is largely what characterizes FM synths, so worth exploring.

✍ *We are mostly going to try to **avoid using the filter**. It's actually really powerful when used in combination with FM, but for our purposes it disguises the sound that FM is making, and over complicates things. So we should "**Keep It Simple, Stupid!"** for now, at least.*

I suggest you make space for a few new programs or prepare to over-write some. By default, the factory banks A and D are identical, so you can chose bank D and write over any programs.

☞ Let's start with a completely blank slate:

1. Turn OFF the **Piano**, **Organ** and **External** sections in both slots **A** & **B**.
2. Turn ON the **Synth** section in both slots **A** & **B**.
3. Turn off ALL **Effects** in slots **A** & **B**, including the master/global **Compressor** and **Reverb** effects, and the **Rotary Speaker** in the **Organ** section.
4. Select slot **A** and press **SHIFT** and **Sound Init** in the **Synth** section. Do the same with slot **B**.
5. In the master section, turn off the **Keyboard Zones** LEDs so we have no keyboard splits.
6. Make sure slot **A** is active and slot **B** is inactive. (if both are active, first press and hold **A**, then press **B** to make it inactive).

7. Save this program – call it "***Init Synth***" or whatever you like.

✍ *We'll be starting from this point quite often, so don't overwrite this program! Save any magical sounds you make along the way in a new program location and keep this "**Init Synth**" program as a handy shortcut.*

Making the first FM program – a Hammond Organ.

I think the best way to approach a new topic is to start in your comfort zone. As you are a NS2 owner, chances are you bought it partly for the Hammond B3 organ emulation. If you understand how that works, then you can apply some of that know-how.

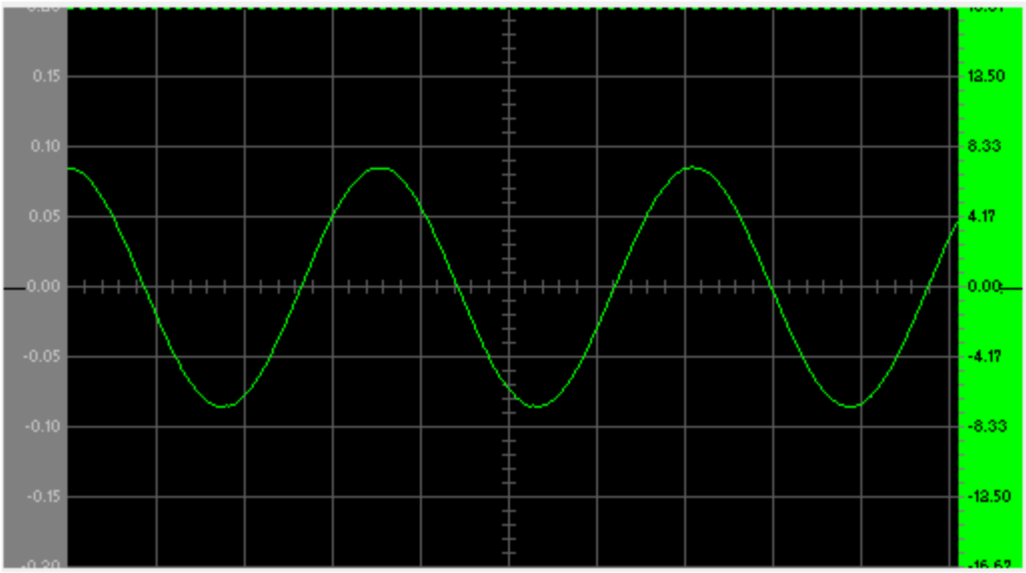
☞ In the **Synth** section press the **Waveform Selector Button** until you have **FM** active, and turn the **Waveform Selector Dial** counter-clockwise until the display reads "**Sin**".

This is what the Synth settings should look like now:

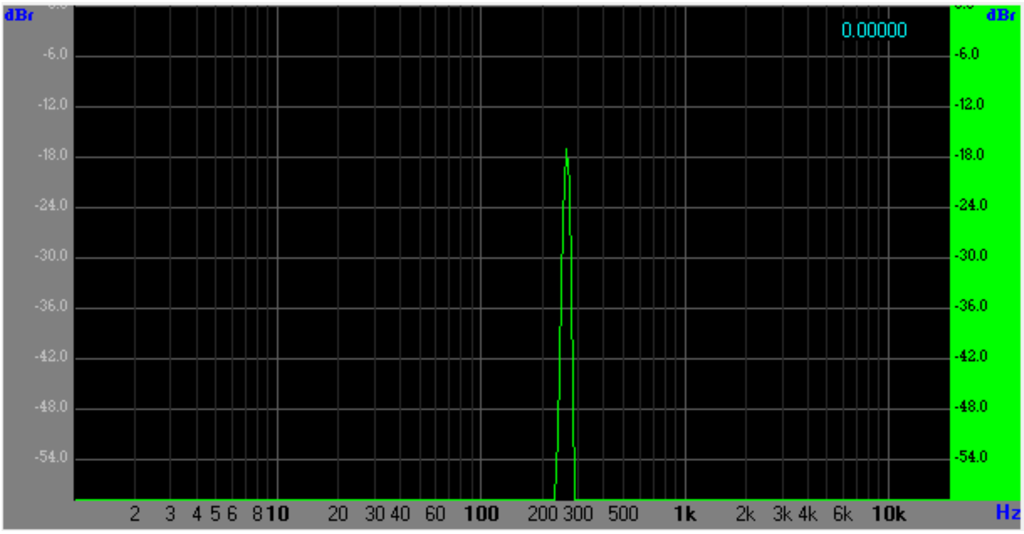
Play a middle C – it sounds about as simple a sound as you can make – and it really is.
"Sin" has nothing to do with straying from the righteous path – it's short for SINE, which produces a pure tone.

What you are hearing is the pure fundamental note C4 with absolutely no overtones. Overtones or harmonics are what you add to give the sound more character.

Here's what the wave looks like on an oscilloscope:



Below are the frequencies involved in the sound. There are no harmonics, just the pure tone at 261.6Hz, which is the frequency for middle C.

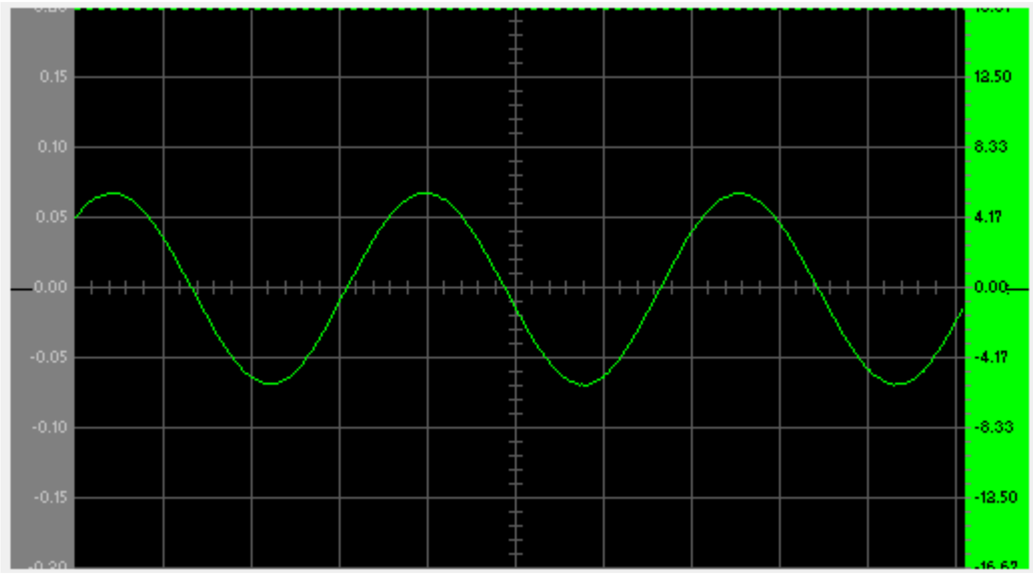


☞ For comparison, switch to slot **B** and turn off everything except the **Organ** section. Choose a Hammond **B3**, and turn off the **Rotary Speaker**, turn off any **Chorus**/**Vibrato** type settings and turn off **Percussion**.
Set the **Drawbars**: **0 0 8 0 0 0 0 0 0**

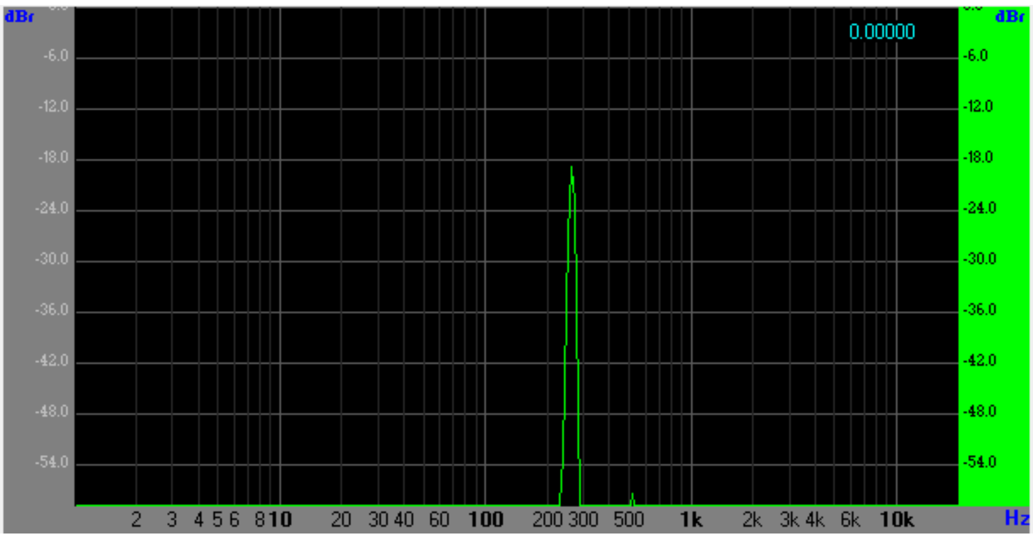
Play a middle C again, and it should sound pretty identical to the FM Synth init sound!

I made an audio file to compare the sounds – it's in the ZIP file, called "**B3 - FM - 008000000.mp3**". The Hammond is first, the FM synth shortly afterwards.

Here's an oscilloscope picture of the B3 waveform…:

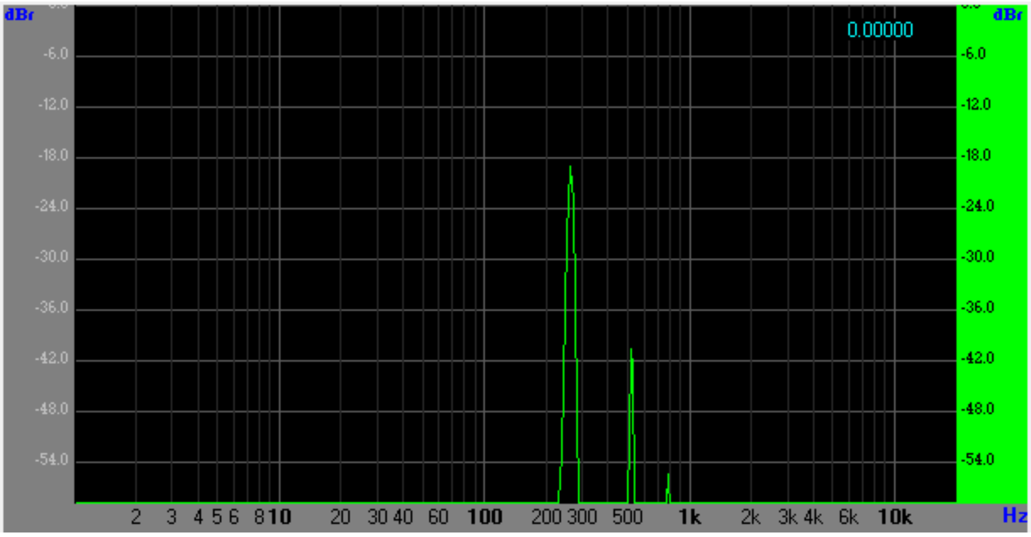


…and frequencies/harmonics (below). Notice there is an additional spike at 522Hz, but it's amplitude is so small it has has only a faint effect on the overall sound (the x-axis is logarithmic, not linear – trust me, it is 522hz!).



☞ Now change the drawbar settings to: **0 0 8 3 1 0 0 0 0**

Here's the picture of the frequencies – you can see the sound is now made up of two main frequencies – one around 261Hz (the fundamental) and one around 522Hz which is the 2nd harmonic for middle C – it's actually an octave above the fundamental.



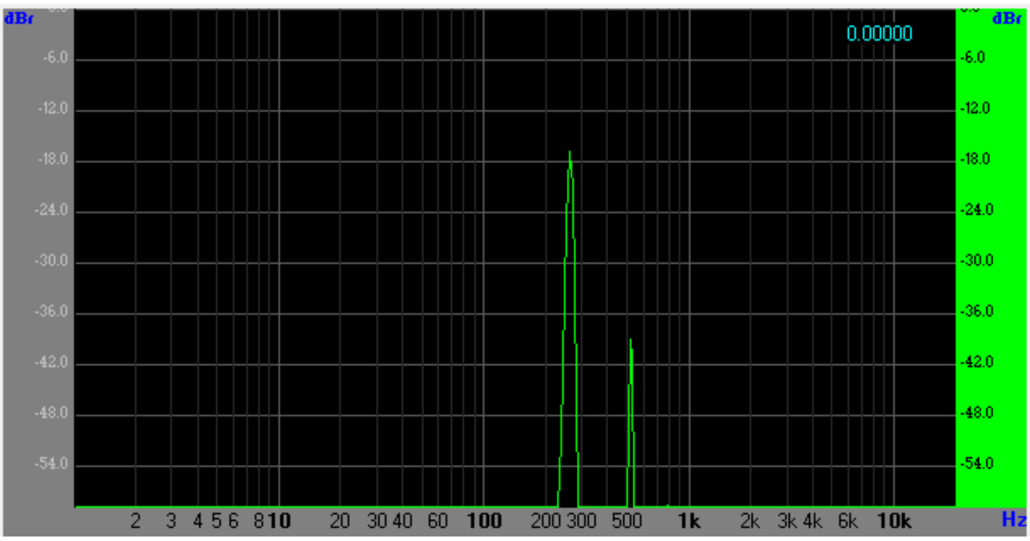
☞ Go back to Slot A where you have your Synth section enabled.

To make this kind of sound on the FM Synth, we need a way to add harmonics to the SINE wave.
*(With 'traditional' analog synthesisers, you use multiple oscillators that produce sounds rich in harmonics, and you use the filter to remove the ones you don't want. That's why it's usually called '**subtractive synthesis'** as you are **subtracting** from, or **filtering** out some of the sound.)*

Anyway, this is FM. To add harmonics, the simplest way is to add some **feedback** to your SINE wave.
☞ Turn the **OSC SHAPE** knob just a bit so the main display reads "**1.4**"

☞ Now switch backwards and forwards between slots **A** and **B**, comparing the 'real' organ, with the one we are making. They sound pretty close don't they?

Here's the frequency chart for the FM Synth:



The sound comparison file is called "**B3 - FM - 008310000.mp3**", once again with the Hammond playing first followed by the FM synth.

☞ Play this FM synth sound through a rotary speaker (set to synth instead of organ), with a bit of drive too.
Not bad for a **single sine wave oscillator**, eh?

Save this program, we'll develop it in a minute after this next techy section:

FM explained a bit more…

The next thing to enhance that sound might be to bring in another oscillator or two, so let's look at that.
But first we have to cover some of the boring sciency stuff. Also I should probably mention the DX7 and how the NS2 does things differently.

In FM, the way that you can generate more interesting sounds than a sine wave as by using one oscillator wave to **modulate** another (see Synth Overview above for a quick summary of modulation). The one that carries the audio signal is called the **carrier**. The one that modulates is called the **modulator.** Whether it's a carrier or a modulator, it goes by the name of an **operator**, for historical reasons known best to Yamaha. An operator is more than just a single oscillator, it also has parameters such as frequency, the amount of effect it has on other operators, the rate at which that effect takes place… more later.

The FM synth comes with a bunch of different algorithms, which are different configurations of carriers and modulators. You access them with the **Wave Selector Dial** under the red LED screen. Some of them involve a modulator modulating another modulator, which modulates the carrier. Some feedback the output wave into the carrier, others don't.
The user manual lists all of these on page 31, but not in a way that is easily understandable if you are new to FM.

The Yamaha DX7 (which was the first generally available FM synth), had 32 algorithms – each was a different arrangement of the DX7's 6 operators.
The NS2 has 38 algorithms, but only 3 operators. Even Yamaha's budget DX synths had 4 operators, so what were Clavia thinking?
The answer is: *avoiding complexity*.
The NS2 is meant to be a tool for musicians to quickly craft the sound they are looking for in a live situation, not go wading through menus.
The way you went about programming a DX7 involved choosing an algorithm, then setting the frequencies, detune, rates, levels for each of those 6 operators, which is not something you did quickly or easily, even if you knew what you were doing.
There was a great deal of flexibility but it came with a steep learning curve.

So where does that leave the NS2? It has much less flexibility to make noises like steam-trains, lazer guns and flying saucers, but Clavia instead appears to have focused on an architecture geared more towards musical tones.
While there are only 3 operators, there are also slots A and B, so you can have 6-operator sounds by layering both slots. There is also the Unison control, which adds between 2 and 4 oscillators for each slot. So that's up to 12 oscillators! (2 slots x 1 carrier osc + 2 modulator oscs + 4 unison oscs), although the 2 modulators aren't audible on their own, they modulate the carrier remember? But that still leaves 10!
You also have a filter which the DX7 didn't have, so you can combine FM tones with analog filtering.
Add the various effects like the rotary speaker, the amp sim, phaser, chorus, delay, reverb etc. and you start to leave the DX7 spluttering in your dust trail!

Algorithms

So lets take a quick look at the algorithms on the NS2 and what they are.
Unlike the DX7, the NS2's algorithms do not allow individually setting the frequency and other parameters of each operator. You can only set the parameters for the carrier. However, this turns out to be not such a bad thing when it comes to frequency: sounds which are 'harmonious' to the ear involve frequencies which are at a specific ratio to the fundamental tone (which is the note you play – e.g. middle C). These harmoniously related tones (to the fundamental) are called **harmonics**.
The NS2 provides algorithms that contain pre-baked ratios of frequencies. If you look at the list of algorithms, you will see a pattern and it becomes less confusing:

- You have always have 1 carrier.
- You can have zero, 1 or 2 modulators.
- The carrier can either feedback into itself, or not
- If there's a dot in the display, the algorithm includes feedback.
- No dot = no feedback (except Sin, where you can disable feedback by turning the OSC Shape knob fully counter-clockwise).

Parameters

As mentioned above, the DX7 is programmed by first selecting an algorithm, then setting various parameters for each operator in turn.
Whilst the NS2 doesn't have this degree of flexibility, it is still possible to make similar sounds to the DX7, once you know the equivalent setting.
The table below lists the main DX7 parameters, the purpose of each, and how to achieve a similar function on the NS2, where possible:

DX7 Feature	Description	NS2 Equivalent
Algorithm	The particular arrangement of operators.	Algorithms can be selected via the OSC

		Waveform selector knob.
Feedback	Sets the amount of feedback to apply – more feedback results in more harmonics and a harsher sound.	Feedback depends on choosing an algorithm which has feedback. The amount of feedback is controlled with the Shape knob.
Operator on/off	Algorithms always have 6 operators. However, each operator can be enabled or disabled.	The number of active operators is determined by the algorithm – 1, 2 or 3. By using both slots A and B, you can have 6 operators.
Frequency Coarse/Fine	The frequency that the operator oscillates at.	There is no way to explicitly set the frequency on the NS2. However, the DX7 featured a frequency mode where the coarse frequency was adjusted in terms of ratio. In the NS2, the ratio is actually built-in to the algorithm you choose.
Detune	Allows the operators frequency to be detuned slightly, which can help fatten the sound. This is caused by the operators to wander in and out of phase, similar to the 'warm' sound of analog synths (which do this naturally due to inherent instability of VCOs – if you have no idea what I'm on about, you should read the subtractive synthesis section of this guide).	Again, there is no way to specify the amount of detune on the NS2, but you can achieve this by using the Unison button. This adds duplicate oscillators which are detuned by increasing amounts depending on the setting. (see page 35 of the NS2 manual)
EG Rate & EG Level	The DX7 has an envelope generator with 4 stages – Attack, Decay, Sustain and Release (ADSR). Each has a Rate of change, and a Level. If the operator is a carrier, this affects the volume, if it's a modulator it affects the timbre.	On the NS2, you use the AMP ENV controls (Attack, Decay, Release) to change the volume over time (the carriers), and the MOD ENV controls (Attack, Decay, Release and Shape Mod) to change the timbre (the modulators). The envelope controls on the NS2 only affect

		the rate of change. (See **Envelopes** section below for more details).
Keyboard Level Scaling (Breakpoint, Curve, Depth)	This allows you to make the EG (volume or timbre) of the operator have a different amount of effect depending on the key played. Each operator has it's own individual settings.	You can vary the effect of timbre change to some extent by using the Keyboard Track button in conjunction with a Filter. The DX7 also used this feature to enable keyboard splits, which the NS2 does anyway.
LFO	The DX7 LFO has parameters for Wave shape, Speed, Delay, Pitch modulation and Amplitude Modulation. An LFO sync feature enables the LFO wave to restart when you press each key. Without sync enabled, the LFO runs continuously in the background and when you press a key it picks up its current position which may be a trough or a peak, or anywhere in between. With a slow LFO especially, this means that the first time you press a key, you may get a "*oo-ee-oo-ee*" sound, but pressing the same key again may give you "*ee-oo-ee-oo*". With sync enabled, each key press sounds identical.	The NS2 lacks a sine wave (for LFO) and only has reverse saw, although a forward saw is not much use in an LFO anyway. Speed is controlled by the LFO Rate knob in the LFO section. Pitch modulation can only be achieved using the dedicated Vibrato button, and is quite limited. Amplitude modulation is achieved via the Shape Mod knob, when turned anti-clockwise towards LFO, but sadly this results in losing envelope modulation (see below). Sync is not implemented in the NS2 at all, which can make slower LFO sounds unpredictable and unusable.
Output Level	This controls the overall depth of the EG – in other words how much effect it has.	This is achieved by turning the Shape Mod knob in the clockwise direction towards MOD ENV. Usually, a middle position of around 5 is the best way to hear the effect it has. Using this means you

		can't use the LFO for modulating the operator (although you can still use it to modulate the filter section).
Pitch EG	This is used to create pitch changes over time using an ADSR envelope just as you do with Amplitude EG to control volume or timbre.	Pitch modulation is only possible via the Vibrato button or the Glide knob*

* It usually doesn't make sense to complain about the lack of certain synth features when you consider the role of the NS2, which is not intended as a synth, but an easy-to-use live performance instrument.
The dual-purpose function of the Shape Mod Env knob for both LFO and envelope is annoying, as is the lack of ADSR envelopes and the limited control of vibrato (although you can adjust the Vibrato Rate, albeit very slightly via the Sound Menu in the main section.)
Generally, the missing features prevent you making more unusual programs like special FX, atonal or dischordant sounds etc., which is not the territory of the NS2.
The lack of LFO sync is the most annoying – if it has to be hard-wired I would expect the NS2 to have sync enabled rather than absent by default.

Back to making sounds

Let's go back to that organ.
So we now know that by choosing another algorithm, we can bring in more oscillators and make the sound richer.

☞ Change the algorithm so the display reads: **1.1**

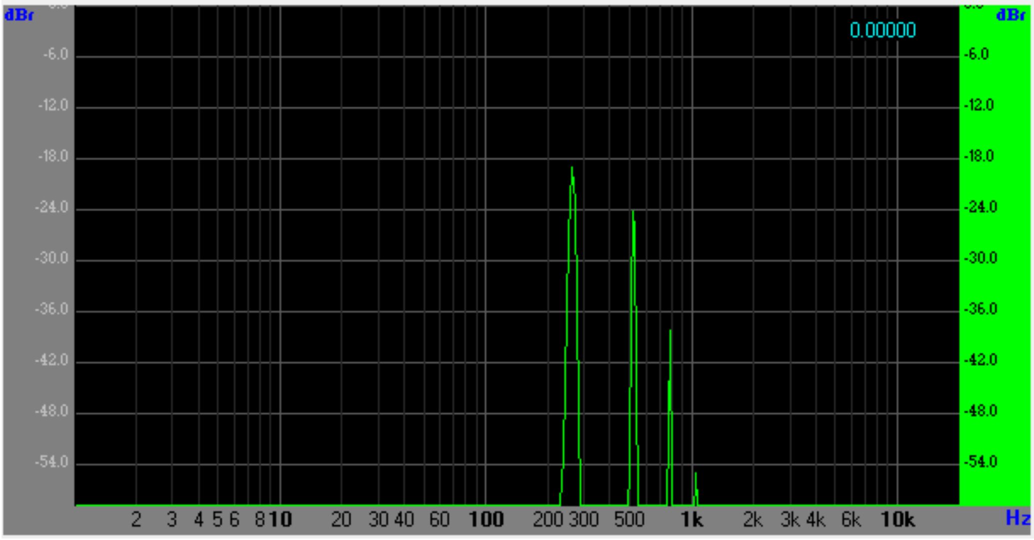
This means the modulator is the same frequency as the carrier, as the ratio is 1:1. The dot also tells us there is feedback control.
This has even more harmonics.

☞ Adjust the **SHAPE** knob down to about **0.9** – generally you don't often need high values on this knob – feedback is usually best in small doses.

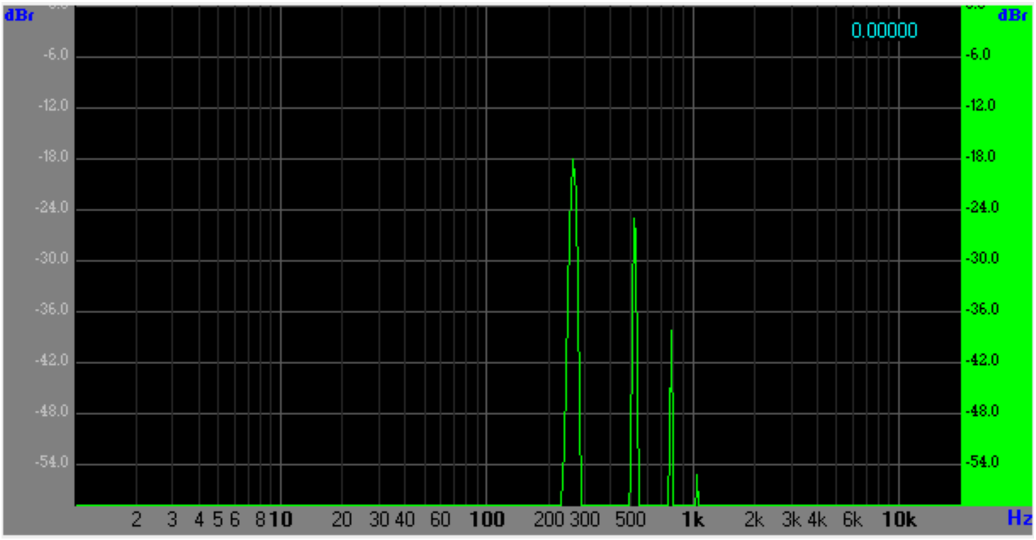
This matches fairly closely a Hammond with the drawbars set to
0 0 8 7 4 1 0 0 0

Once again, a couple of harmonics charts to compare:

The Hammond:



...and the FM Synth:



☞ Play this through the **Rotary Speaker** and it sounds pretty good.

Another way to fatten up the sound is to set the **Unison** control to 1 or 2 – this adds another oscillator that is a slightly detuned copy of the carrier. This was a common trick in DX7 programming, except you had to do it tediously by hand. On the NS2 you just press 1 button!

The next algorithm to give us another operator and also feedback is **1.11**
By now, the combined effect of two modulators has taken us away from organ territory, so you can see, sometimes less is more (the same goes for unison).

Incidentally, if you wanted to add harmonics *below* the fundamental frequency, you can on the Hammond with drawbars 1 and 2, but you can't with the FM synth. However, you could add another FM synth in slot **B**, but transposed down an octave, then play both slots together.

The last thing we are going to do with this sound before moving on is to reproduce that percussive click you get with electric organs.
I think this is caused by a momentary surge of voltage when the key is first pressed (*please correct me if I'm wrong!).* Anyway, we can reproduce this by means of the Modulation Envelope (**Mod Env**) – see the overview to the synth section above if you are hazy).
The **Mod Env** allows us to vary the way the Modulator affects the carrier.

☞ Set the **MOD ENV DECAY** to about 12ms. Now turn the **SHAPE MOD ENV** knob fully clockwise. If you play a note, there's a hefty click at the beginning.

This is because the modulator and the feedback are initially applied as though you had the **SHAPE** knob set to the full amount. However, the **Mod Env** has protected your ears by dropping it almost immediately (over 12ms) to the value you set it at – about 1.4. We need this to be less dramatic, so instead of starting with the full amount of modulator/feedback, we just want a slight peak above the level we set:
☞ turn the **SHAPE MOD ENV** knob back to **2.0**.

Now the sound starts with a modulator/feedback level of **2.0** and drops to **1.4** in 12ms.

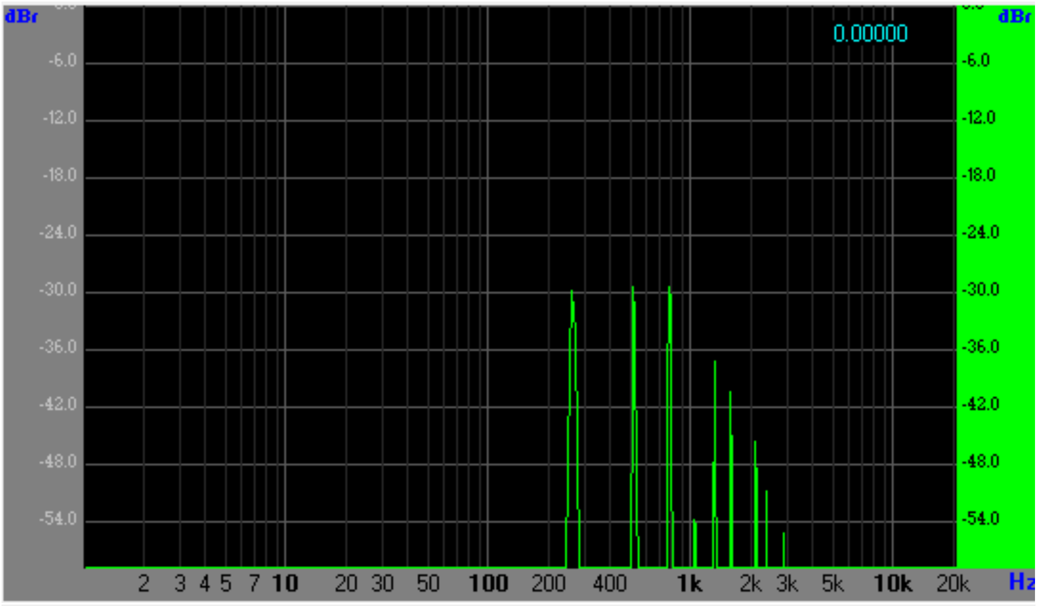
That's it with the organ – we've covered quite a lot of ideas.
To be honest, you already have a top-of-the-range Hammond organ simulation in the NS2, so you probably won't be needing this FM organ much, but it was a good starting point and allowed for easy comparison.
Let's continue that theme by creating an electric piano sound – the DX7 became famous partly because of its distinctive EPiano1 patch, which appears everywhere in 80's music.

The secret of Envelopes

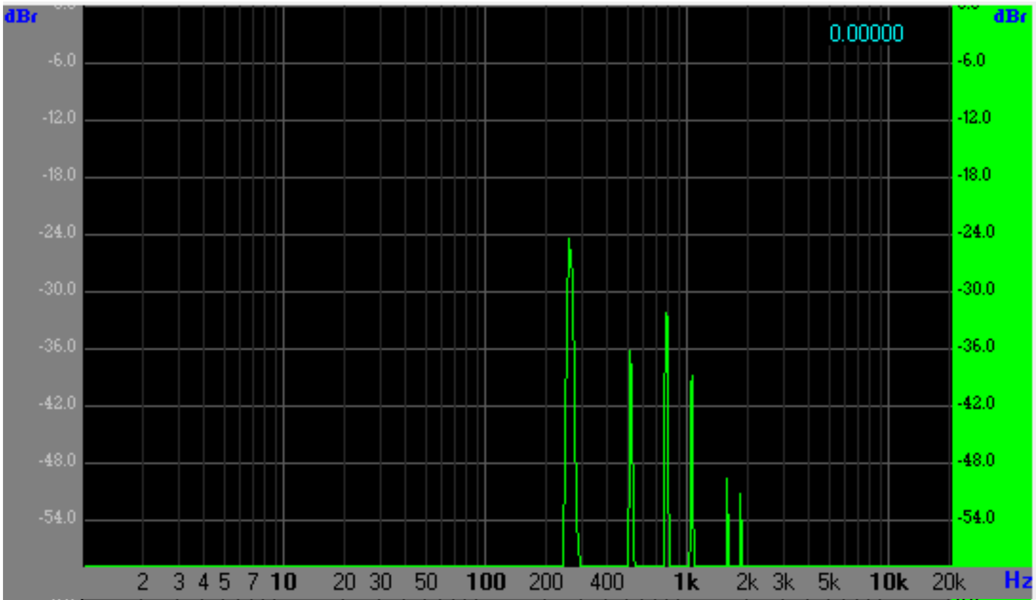
The organ sustains its sounds indefinitely (as long as the key is pressed), but most instruments don't. Percussive instruments like the piano start to make sound when the string is hit, then the sound changes and dies away as the string comes to a rest. Wind/brass instruments can sustain much longer, but eventually the player runs out of breath. Even with stringed instruments, where the bow can continue to make the strings resonate, the sound is affected when the bow changes direction.
Just to recap - this change in sound over time is controlled by an **envelope**,

Envelope modulation is the key to making great FM sounds, regardless of whether you are making an emulation of an acoustic instrument, or a completely original sound.
With the organ sound, we used the Mod Env to add a more percussive click at the beginning of the sound, but with a piano sound we are going to use it to vary the amount of harmonics over time.

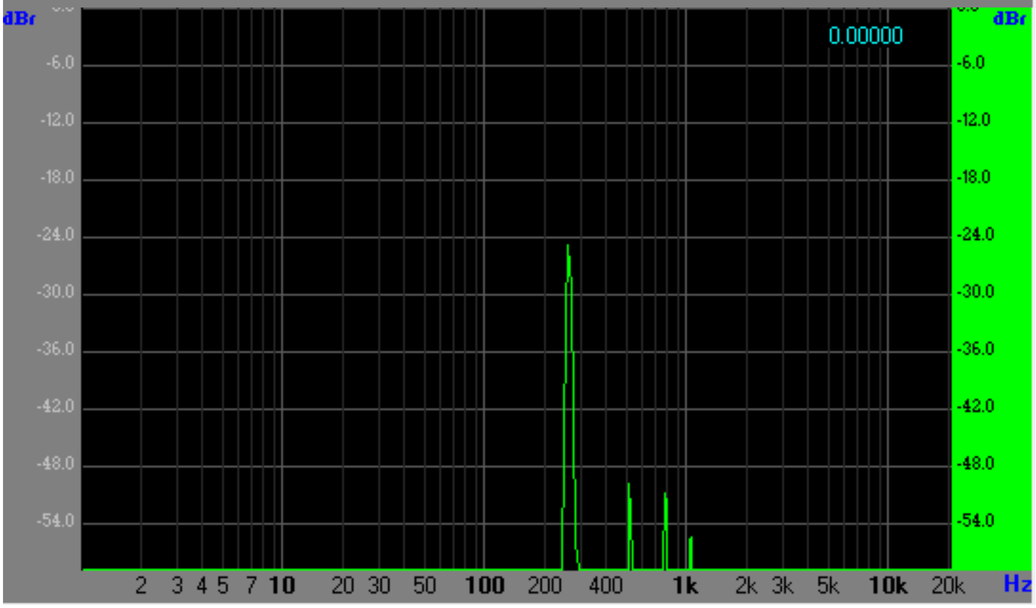
The following page shows a timelapse of the harmonic frequencies of a Rhodes piano played via the NS2's piano section (EPiano 4 "Clos Ideal"), once again playing middle C.
The first chart is immediately after the note is played, the second is a about 200ms later, and the third after about a second has passed.



dBr
0.00000
-6.0
-12.0
-18.0
-24.0
-30.0
-36.0
-42.0
-48.0
-54.0
2 3 4 5 7 10 20 30 50 100 200 400 1k 2k 3k 5k 10k 20k
Hz



dBr
0.00000
-6.0
-12.0
-18.0
-24.0
-30.0
-36.0
-42.0
-48.0
-54.0
2 3 4 5 7 10 20 30 50 100 200 400 1k 2k 3k 5k 10k 20k
Hz



dBr
0.00000
-6.0
-12.0
-18.0
-24.0
-30.0
-36.0
-42.0
-48.0
-54.0
2 3 4 5 7 10 20 30 50 100 200 400 1k 2k 3k 5k 10k 20k
Hz

You can see that when the sound starts, there are lots of harmonics. After a fraction of a second, several of these die away, and over the next few seconds, more die away at different rates, leaving just a few dominant ones in addition to the fundamental frequency.

Making a piano sound

To emulate the sound, we'll look at these dominant harmonics (picture 2) and find the closest match in the FM algorithms.

☞ Find a new empty program, copy your **Init Synth** program into it.

☞ Make sure the FM synth is selected, and turn the **SHAPE** knob a fraction up from zero, to a value of about **0.6**, any more than this, and you won't be able to find a suitable algorithm as there will be too many unwanted harmonics.

☞ A good starting point for a piano sound is algorithm **3.1**, so select that**.** This provides some but not all of the harmonics. Once you get the basic piano sound in slot A, you can create another in slot B with a different algorithm such as **21**, which adds some of the missing frequencies.

If you come from a DX7 background, don't fixate on the number of operators in the algorithm – don't forget we can add more using Unison later.

It's completely up to you what order you go about crafting a sound, but having selected a basic oscillator waveform (or algorithm in the case of FM), the next logical step is the amplitude envelope.

☞ The piano is percussive, which means the sound starts almost instantly. Set the **AMP ENV ATTACK** to its zero position (**0.5ms**). Once the string is struck, it will slowly continue to vibrate while the key is down and the hammer is off the strings. Set the **AMP ENV DECAY** to a knob position around 6 (**3.42s**), which seems about right. When the key is released, the hammer sits back on the string again, causing the strings to stop vibrating fairly quickly, but not instantly. Set the **AMP ENV RELEASE** to just past 3 (**265ms**).

Let's face it - it's never going to sound like a real piano, but at least it now has the right kind of volume characteristics.
The next step is to do something about the character of the sound over time, so we turn to the modulation envelope.

☞ Turn the **SHAPE** knob to a position where about three LEDs are lit (**1.5**). This sets the minimum amount of modulation/feedback.

☞ Turn the **SHAPE MOD** knob clockwise towards **MOD ENV**. This sets the degree to which the modulation envelope will affect the modulator/feedback. A value of about **6.1** is perfect.

☞ Now set the **MOD ENV ATTACK** to zero (**0.5ms**), **MOD ENV DECAY** to about 5 (**1.39s**), and **MOD ENV RELEASE** to 4 (**851ms**).

Hopefully by now you can start to see how this is working: we start with a more harmonically rich sound, and then move to a more mellow sound for the rest of the decay time. The way the harmonics fade away happens faster than the way the volume drops away, which is why the **MOD ENV DECAY** is a smaller value than the **AMP ENV DECAY**.
Why is the **MOD ENV RELEASE** higher than the **AMP ENV RELEASE**? This results in staccato notes having a brighter release than notes with a longer duration.
I looked at the hammers on my piano to figure out why, and I noticed that when you hit the key and release immediately, the sound is harmonically brighter than when you play a longer note and then release. In the first case, the string is damped while it's resonating more quickly than in the second case, so that may explain it. Anyway, sometimes you do things just because they sound right.

By now, we have a pretty good electric piano. Let's make it as good as we possibly can.
We can simulate more strings by adding more operators.

☞ Press the **UNISON** button until both '**multi**' and '**1**' are lit. This adds some duplicate operators, slightly detuned from the 'main' one.

Now to add some expression capabilities.:

☞ Enable **AMP ENV VELOCITY**. This means that the harder you hit a key, the louder it will be, whereas soft playing will be quieter. The Italians would say piano/forte.
We want hard notes to be brighter, and softly played notes to be mellower:

☞ Enable **MOD ENV VELOCITY**. Now, in addition to harder keypresses being louder, they also have more modulation envelope applied. In our piano program, this means more harmonics and a brighter sound and fewer harmonics and a mellower sound when played softly. To make the most of this effect, I would now turn the **SHAPE** right back down to its zero position, as we are effectively now using velocity to turn the knob. How much it is affected by velocity is governed by the amount of **SHAPE MOD ENV**.

If you haven't already, save the program into a new location.
Additionally, you can store your newly created sound in the synth memory area alongside the sampled sounds (and preset synths) - then you can use it again in other programs. Check page 35 of the NS2 User Manual.

Now you have a basic 1-slot ePiano sound.
A healthy amount of Reverb from the FX section would be appropriate.

From here, you can go where you want: you can make a much richer sound by copying Slot A to Slot B (*see note below), and then tweaking the Slot B piano. Don't forget to enable both Slots A and B to get the fully layered sound.

For Slot B, you want a different algorithm that compliments Slot A by filling in some of the missing harmonic frequencies. I have included a few example ePiano sounds in the accompanying ZIP file. One uses algorithm **3.1** with **221**, another uses **31** with **1.1**.
It's also worth giving each Slot a different MOD ENV character – one can be percussive and bright, decaying quite quickly, while the other can be more mellow and decay more slowly.

Try adding a Morph to control the amount of shape with the mod wheel.

You could also try making Slot B into a more bell-like sound by using some of the higher ratio algorithms.

✍ *A hidden secret of the NS2 is copying the sound in 1 slot to another, which is incredibly useful: if you want to load a sound into Slot B only, without affecting Slot A, hold down the Slot B button, then turn the Program Selector knob until you see the Program and the slot where you want the sound to come* ***from****.*
If you want to copy the current Slot A into Slot B, do the same, but turn the Program Selector knob just 1 'click' so it shows Slot A of the current program.

It's all about the bass

The next sound we'll go for is an FM Bass. There'll be less talk of harmonics and other tedious stuff – by now you should get the idea.
The sound I have in mind is something like the bass used in Kenny Loggins' TopGun theme – pretty close to one of the DX7 factory presets.

This sound will continue the idea of layering both A and B slots to contribute different elements of the sound. This means you are essentially adding an additional operator in a parallel configuration.

✍ *DX people – this may be a little like ALG 3 where the left branch has 3 ops and no feedback, and the right branch has 3 ops and feedback on the last modulator. In the NS2, Unison would add an additional branch, e.g. ALG 28. Anyway, don't get too hung up on algorithms – read the comparison table above if you haven't already, and just accept that the FM 'palette' of the NS2 is not as colorful as that of the DX7.*

☞ Copy your **Init Synth** program, to a new empty location. A good starting algorithm for a bass is one with a low ratio, so choose the **111** algorithm.

☞ As before, set the **AMP ENV** first. **ATTACK** = **0.5ms**, **DECAY** = **1.05s**, **RELEASE** = **120ms**.

☞ Crank the **SHAPE** knob up to **1.9**, and turn the **SHAPE MOD** knob towards **MOD ENV** to a value of **5.1** (about halfway).

☞ Now the **MOD ENV**. Set **ATTACK** = **0.5ms**, **DECAY** = **206ms**, **RELEASE** = **206ms**

☞ It sounds like a bass alright, but a bit weak. Fatten it up by setting **UNISON** to **1 + Multi**

Now to give it more character by layering another element to the sound. Slot A is the 'pluck' while Slot B will provide some of the bassy string resonance.

☞ Copy **Slot A** to **Slot B** (see the note on the previous page for how to do this).

☞ Make sure only **Slot B** is the active one, and not both slots. You can quickly get confused by tweaking parameters on the wrong slot while both are active!

☞ Change the algorithm to **211**. It's got some good characteristics, but we want lower frequencies so drop it down an octave using **OCTAVE SHIFT** = -1

☞ Set the **AMP ENV** to provide a slower decay and release: **ATTACK** = **0.5ms**, **DECAY** = **2.85s**, **RELEASE** = **588ms**.

☞ Turn the **SHAPE** knob up to **3.0**, and turn the **SHAPE MOD** knob up a little (still towards **MOD ENV**) to a value of **6.6.**

You can leave the **MOD ENV** the same as slot A.

Enable both slots and play some 80's bass-lines!

More things to try

- You can fatten the sound even further using **FX – CHORUS** on each slot.
- How about some slap-back **DELAY?**
- Try changing the modulation characteristics of one of the slots, by changing the **SHAPE** and **SHAPE MOD** knobs
- Set the **LFO RATE** knob to about **6.5hz** and the **LFO waveshape** to a reverse saw (the bottom shape). Now turn the **SHAPE MOD** knob anti-clockwise about halfway towards **LFO** position.

Next up – some nice metallic sounds, which is what FM is really good at.

Coming soon...

Wavetable and single-cycle waveforms

To do…

Tips and tricks

To do…