

# MeloDramatik

Version 1.0.0

Operation Manual



# ENLIGHTENSPEED

# Table of Contents

Introduction	5
The Main Concepts	7
RELAX!!!	7
The Melodic Inversion Algorithms	8
Notes on context of the intervals	9
The Double Interval Algorithms	10
The Asymmetric Interval Algorithms	10
The Spread Algorithms	10
Random	11
Reset	11
The differences between A, B, and C etc.	11
The Combination Algorithms	12
Notes on monophonic and polyphonic input	13
The Global Areas	14
Sequencer, Step and Global Modes	14
The Step Trigger	14
Global Algorithm	15
Input Filters	15
Output Filters	15
Output Filter Rules	16
Patterns	16
Pattern Functions	16
The Patch Area	17

Mute/Pass Global Controls	17
The Sequencer Panel	18
The Algorithm Channels and Step Types	18
Draw-X Mode	19
Steps	19
Rate	20
The Randomize Button	20
Mutes	20
Octave Shifts	21
Passthroughs	21
Semitones	21
Scales (Sequencing)	22
Spreads	22
The Edit Panel	23
Randomizer Settings	23
Scales	23
The included patches	24
Using with other Players	25
Routing	25
Using Direct Record	26
Acknowledgements	28



# Introduction

Hello and welcome to MeloDramatik from Enlightenspeed!

You now have in your possession one of the most advanced song writing and remixing tools in all the galaxy. MeloDramatik can take existing MIDI files and give them a new lease of life or be used as a MIDI remixing tool for advanced groove samplers such as Serato Sample.

This manual should help you understand the concepts and operation, but the only way to truly know its power is to use it.

## **A little history**

This is effectively the third evolution of this product line, that started with the original incarnation of Note Alter back in 2019. Note Alter simply randomized all the notes that came in and spat them out as something else, which was great for drums but harsh for melodic material. Thus, the original version of Melodic Inverter was released to give a seemingly random but actually very mathematically sound function for getting more predictable results from the inputs. That design was painfully simplistic and didn't even have a real GUI, so was further evolved into the later version of Melodic Inverter which added the step sequencer and the randomizers etc. and completely reinvented the whole concept of inversion and pattern mutation.

The second version of Melodic Inverter had two weaknesses, firstly it was now far more than just an Inverter, so the name was less appropriate, but as it had been developed in conjunction with the fine people of the ReasonTalk.com community and gained arms and legs with a barrage of user requested features, the design, in terms of both GUI and software had become very inflexible, very quickly as new requests were bolted on to original parts.

Thus, the concept for MeloDramatik was born, with many new headline features and many simplifications included in a redesigned GUI shell and with all code completely rewritten from the ground up. There are still some

similarities to the general look and feel of Melodic Inverter, but it is best thought of as an entirely different program.

So, what you now see when you look at MeloDramatik is very much a “perfected” version of little old Note Alter – which is itself still great, of course, but just not this great!

### **What does it do?**

First up it can perform melodic/chromatic inversion as you might expect given the history, but it can now do this in multiple ways and also use other algorithms such as double interval, spreading, asymmetric interval, static combinations and random.

These algorithms are either accessed globally, or they can be sequenced over time to give uniquely complex “evolving” combination algorithms. Using the randomize functions lets you access these very quickly and the number of possible combinations available is more than anyone could ever use – when we tried to work it out, we gave up at the point when it passed 1 trillion – and it was still a long way from finishing! Using fuzzy logic tells us that it is at least  $x \cdot 10^{30}$ , where  $x$  is in the range 1-10. You won't run out.

You can store up to 8 patterns, and these have supplementary features such as Copy and Paste.

We have also added the filtering section which includes wraparound and foldback handling modes as well as classic filter termination.

There is also a range of complementary features to aid in sequencing such as Mute, Pass, Octave and Semitones all being controllable by step, and the Scale sequencing lane which can also be a powerful standalone tool.

There are further complementary features for editing such as various reset and group controllers, and the Randomize section is deeply customizable, including range controllers and individual triggers. Add to this that you can now draw quickly along the x-axis when the mode is engaged, making it considerably faster and easier to edit parameters.

# **The Main Concepts**

First of all,

**RELAX!!!**

**When faced with a new device of this level of complexity it is easy to feel overwhelmed with the number of possibilities. This device was designed to make billions of possible operations simple and intuitive to achieve, and with exceptionally little requirement for the user to really do that much.**

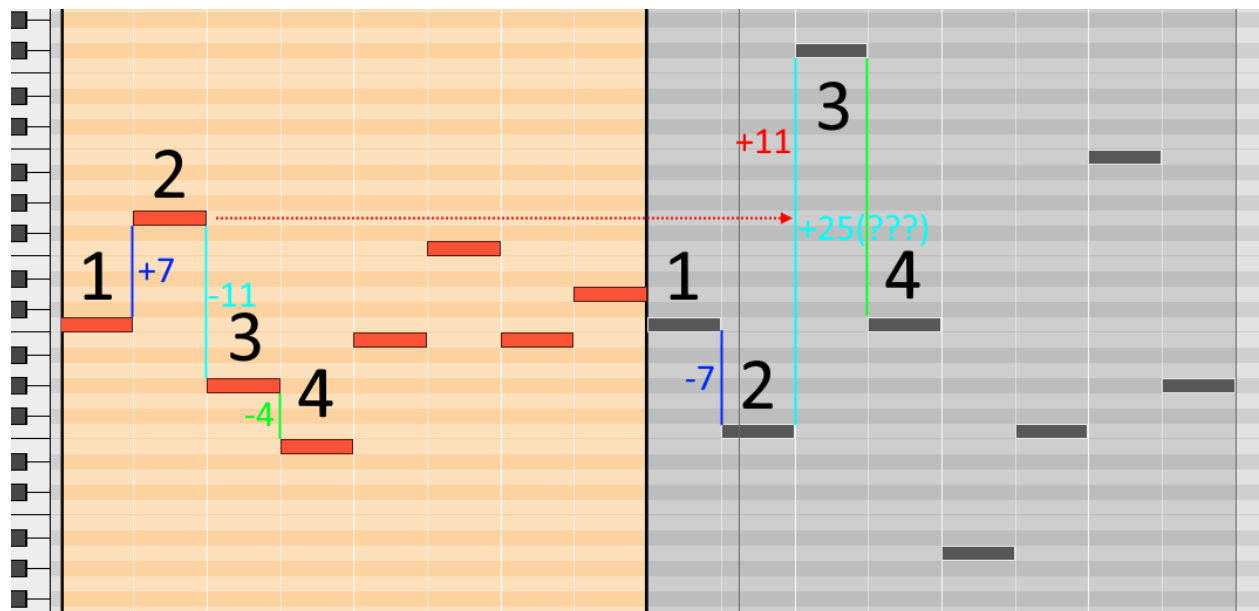
**It is therefore recommended that you familiarise yourself with the basics and then just crack on – you don't need to understand all the ins and outs of how this device works – you can think of it as a magic box while you hit Randomize All repeatedly and be none the worse. It would be nearly impossible to accurately predict the outcome of any relatively complex input processed with even the simplest settings of MeloDramatik. Don't worry if you don't understand what all the algorithms do, it really isn't that important for using the device effectively.**

## The Melodic Inversion Algorithms

The idea that kicked it all off.

Melodic inversion is incredibly simple, yet stunningly powerful. At its simplest it is a mathematical process applied to existing note data, but the result is profound. For the following examples we will only use monophonic material to demonstrate the effects more clearly, then at the end of this section we shall discuss the effects on polyphonics.

First up, let's look at a simple monophonic MIDI segment.



The notes above, taken from the Reason sequencer using Melodic Inverter A show the differences between the input in the orange section on the left, and the Melodic Inverter A output on the right in grey.

The change between the 1<sup>st</sup> and 2<sup>nd</sup> notes shows a nice easy to understand process. Note 2 is originally on G, 7 steps up from Note 1 on C, and on the output it is now on the F, 7 steps down from the original C. Simple.

The change between the 2<sup>nd</sup> and 3<sup>rd</sup> notes is far more radical. In the original version we see that the interval is 11 steps down, but the output shows a



gap of +25 steps! What is happening is that 3<sup>rd</sup> note is actually +11 steps from the original placement of the 2<sup>nd</sup> note. The 2<sup>nd</sup> note has been shifted down by 7 though instead of up, so the resulting interval is thus:  
 $7+7+11=25$ .

As an exercise, do the maths in your head for the interval between the 3<sup>rd</sup> and 4<sup>th</sup> notes, and if necessary, continue until the end of the sequence.

## Notes on context of the intervals

Before moving on we briefly need to discuss some other points of interest. Firstly, the opening note. Many users of the original GUI-less version of Melodic Inverter have asked about the first note being no different on the 1<sup>st</sup> pass, and then different on the 2<sup>nd</sup> pass if looped. The inversion algorithm, and indeed any algorithm anywhere, needs an initial state to start from and then go to work. Because the first note of any sequence has no predecessor, there is no interval for it to be inverted from. In a looped scenario, because the device works in realtime, this interval is generated by the distance between the **final** note of the loop on its previous pass and the first note, which is now no longer the first note. This means that it's usually a good idea to have a listen to both 1<sup>st</sup> and 2<sup>nd</sup> pass versions, to determine whether you prefer one or the other, or both.

Next, it shouldn't really come as any surprise when looking at the 2<sup>nd</sup> to 3<sup>rd</sup> note interval, that larger original intervals can really move things about a bit. In this case we decided that any notes which are thrown right out of the standard MIDI range of 0-127 are cancelled. Unless you are using a sampler patch where the samples bear no relation to the MIDI pitch, then this should probably come as a pleasant surprise. MeloDramatik now has multiple filtering options for this including changing the ranges. See the section on filtering for more details.

Lastly, running consecutively increasing or decreasing pitches will yield considerably fewer radical results. The output will "level out" faster, and the

result will be mostly the same intervals just pitched up or down a bit. Mathematically this is exactly what should happen. If you want to increase the effect, just go into the sequencer and grab a few notes at random and move them up and down.

## **The Double Interval Algorithms**

Now that we have some basics, let's look at an even more effective method for altering the original that doesn't even need a diagram! Simply using the basic concepts above, instead of inverting an interval it doubles it. So instead of an interval being +5, it becomes +10.

This is very simple of course, but the important thing to note here is just how quickly things can move out of the MIDI range, or indeed just out of the realistic range for a particular instrument or musical arrangement. For cases where there are problematic results you should use the onboard filtering.

## **The Asymmetric Interval Algorithms**

These are the same as the double interval algorithms but with asymmetric processing. There is a positive and a negative asymmetric version. See the section regarding the Differences between A, B and C for more details on this.

## **The Spread Algorithms**

The 4 Spread algorithms give a user definable version of the Double Interval whereby if the user inputs notes with intervals of +4, +6 and -4, then the output, when the Spread control is set to +3, will be +7, +9, and -7.

This gives a more controlled and easier to manage approach than the other algorithms.

## **Random**

We decided that with the random algorithm on this product, to make it a little tamer than it is on, for example, Note Alter. In this case the Random function is set to choose a random note within one octave up or down from the original note.

## **Reset**

The reset algorithm is there simply to allow the user to take back a little control of sequences which are getting messier. When a reset function is called it will pass the incoming notes through untouched, and also reset the holding variable which tells the other algorithms the interval of the previous note. This is particularly of use in the Sequencer and step modes.

## **The differences between A, B, and C etc.**

As can be seen, small differences in the mathematical formulas can have big differences in the results. In the original version of Melodic Inverter we decided to use a proprietary algorithm to try and get control over polyphonic material, with modestly successful results.

Over time we received more and more user requests to add the classic “Rachmaninoff algorithm” to our list. The differences between these algorithms are subtle; the ‘A’ version algorithms select their new positions using the last note that the device encountered BEFORE processing, while the ‘B’ versions use the last note AFTER processing.

The Rachmaninoff version of the technique goes a little further, by using the post-processed version and then using it as the positional in the algorithm, thus giving that nice balanced look and feel. Thus, we have Melodic Inverter C, which is this classical technique.

## **IMPORTANT**

In order to achieve the same results as classical composers using the above algorithms, it is vital that you work chromatically and without filtering, as the post processing numbers are affected by these processes.

### Spreads A/B

You may be delighted to hear that this is much simpler.

The Spread A and B versions are simply there to fulfil legacy requirements. Spread A is the full range of -24 to +24, while B is only from 0 to +12 (as it was originally in the second version of Melodic Inverter). To facilitate this within the sequencer and randomizer, there is some maths applied to the input whereby if the Spread is set to B the range is halved, and negative values given their absolute value. So, if the value shows as -17 and the algorithm is Spread B then the actual value used is +8; all half values are rounded down.

## **The Combination Algorithms**

There are now multiple combination algorithms, but each is based on the simple concept of “Algorithm X combined with Spread X”.

These work mostly as you would expect, with the Spread values being added to the algorithm outputs, respectively. As there are four different Spread controllers, you get 4 of each one with some duplexed using the A & B methods, linked to the appropriate Spread parameter value.

So, in total, from a single monophonic input segment, you can get an extra 36 arrangements of the same source which are all reproducible without bouncing down the MIDI and manually adjusting things. We didn't actually add in every single possible permutation, as it felt like overkill.

If you desire you can either abandon the original key and scale by setting the Scale section to be fully chromatic or quantize them all to the same scale; but we're sure you will agree that despite all the weird maths stuff going on above, the output results are almost always magnificent.

### **Notes on monophonic and polyphonic input**

A final word on this before we talk about setting up the device itself.

The classical techniques employed by various baroque and romantic era composers were almost exclusively monophonic. This is only common sense really because monophonic sources, at least would not require a whole new set of complex rules regarding how they were to be inverted or otherwise altered.

We decided that we wanted polyphonic anyway because we are letting a machine do all the heavy mathematical lifting. :D

The results are considerably less predictable, and far more likely to move around all over the place, more so when using "open chords".

Again, note filtering may be the best solution here if things are getting out of hand. Our onboard filters default to on because of this.

# The Global Areas

## The Sequencer, Step and Global Modes

Control of the general mode of the device is governed by the 3-way switch shown below.



The Sequencer and Step modes are governed by the parameters on the Sequencer panel, and the Global Mode is governed by the Global Algorithm parameter.

Each mode has its own merits, but in general, the Step and Global modes are quite cognitive and deliberate in usage, whilst the Sequencer mode, particularly when randomized is an experimental free for all. That said the Sequencer can be the most deliberate of all if using it as a controller with a slow step rate.

The 3-way switch is always present on the main panel of the device regardless of the display mode.

### The Step Trigger

To the immediate right of the mode switch on the panel is the Step Trigger.



This simply controls what Step the sequencer is on when the device is in Step mode. In Global or Sequencer mode it is ineffective. This is, of course, Remoteable.

## Global Algorithm



The Global Algorithm is how you change the algorithm when in Global Mode. Of course, you can use this as an on-the-fly controller for the whole device, but another great usage for it is to help you fine tune the individual algorithms, particularly the Spreads – then once this is done you can go to town with the Randomizer on the Sequencer.

## Input Filters



There two sets of filters onboard the device, the first is a High Pass and Low Pass pair for the inputs on the device. Any note number lower than the value of the High Pass is removed, as is any number higher than the value of the Low Pass.

## Output Filters



The second set of filters is a High Pass and Low Pass pair on the output, which will only work in the same way if the output mode is set to 'T'. Otherwise, the reaction will be as specified below.

## Output Filter Rules

The three output filter modes 'F', 'T' and 'W' stand for Foldback, Terminate and Wraparound, respectively.

In Foldback mode, any value which crosses the filter threshold will be bounced back by the same amount which it crosses the threshold. With an LP filter threshold of 90 and a note number of 98, the output will be 82; not including scaling.

In terminate mode the filter is applied immediately.

In wraparound mode the value is placed at the other end of the scale and allowed to continue. With an LP filter threshold of 90 an HP threshold of 40 and a note number of 98, the output will be 48; not including scaling.

If the range between thresholds is quite tight and the note number either folds or wraps but then passes the opposite threshold then the event will be terminated.

## Patterns

There are now 8 patterns that you can use in the device.



These work like normal patterns in any other device, but at this stage they do not correspond to the sequencer pattern lanes. This is because of a longer-term strategy.

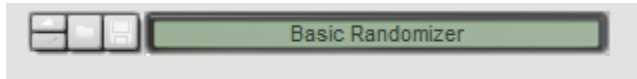
## Pattern Functions

As seen above, there are also standard functions available to the Pattern section: Copy, Paste and Clear, which work as you would expect.



## The Patch Area

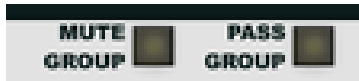
MeloDramatik uses patches, and the design scheme is the same as all other Rack Extension devices.



Please see the Reason user manual for further details.

## Mute/Pass Global Controls

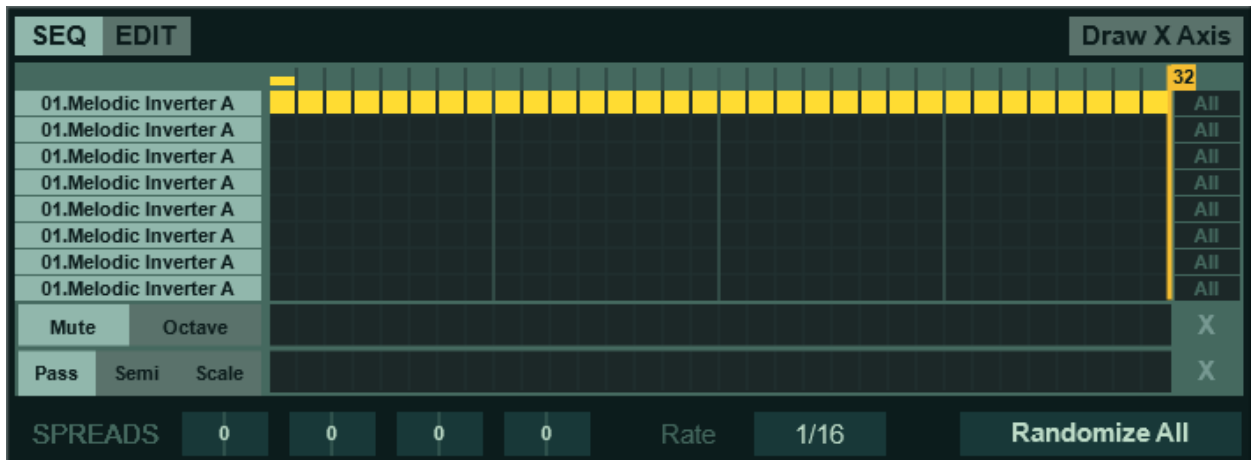
Lastly, there are controls for the Mute and Pass lanes, which switch them on and off as a group, and are Remoteable.



# The Sequencer Panel

The device has two major modes of display, compressed into hide-able panels, like Thor's oscillators if you've never used anything else which does this.

We have the Sequencer panel...



... and the Edit panel (see Edit Panel section).

Each of the panels is accessed by the Seq/Edit buttons and hides the other to save space – the Seq/Edit button has no effect on the input and output of the device. We'll discuss the rest of the panel details in their individual sections.

## The Algorithm Channels and Step Types

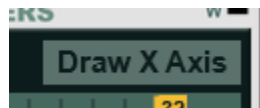


There are now eight algorithm channels in the Sequencer panel: 1, 2, 3, 4, 5, 6, 7 and... I can't remember the last one. These are on the left-hand side of the image above and can be assigned to any of the 53 algorithms.

Each step has its own fader, with channels 1-8 available as steps. The basis of operation is pretty much the same as any other sequencer, where steps are controlled by the timeline, the Step Type governs which of the eight selected algorithms is in use, and when notes come in they are assigned to an algorithm as applicable.

## The Draw X Axis Mode

To make things easier we added the Draw X Axis mode for editing the parameters more quickly.



When active the mode allows you to draw long sweeps into each of the stepped lanes, including the octaves and semitones etc.

Leaving the mode inactive is better for more deliberate editing.

## Steps

Parameters on the sequencer panel are governed by the step number, which is indicated by the timeline bar along the top - this is very similar to many other sequencer devices.

When using the device in Step Mode, the step on the timeline is only changed by the Step Trigger, and in Global Mode it is not relevant.



The Steps controller determines the number of Steps before the Sequencer loops back to the start. The range is any integer between 1 and 32 inclusive. Any power of two (2, 4, 8, 16, 32) is safe for predictable looping, but the more adventurous settings are the higher odd numbers, which can give a constantly evolving set of outputs.

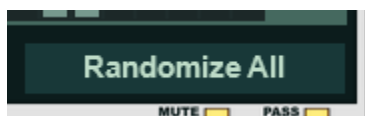
## Rate



The Rate controller specifies how often the step will change when the sequencer is running, and it has the following values available:

8/1, 4/1, 3/1, 2/1, 7/4, 6/4, 5/4, 1/1, 3/4, 1/2, 3/8, 1/4, 3/16, 1/8, 1/16D, 1/8T, 1/16, 1/16T, 1/32, 1/32T, 1/64, 1/128.

## The Randomize Button



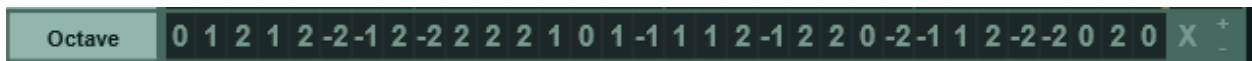
This is simply a trigger button to tell all the selected elements to Randomize. This is placed on the Sequencer page to provide immediate visual feedback for large operations. For more details on this see the Randomizer Settings section.

## Mutes



Each Step also has its own Mute button, and as would be expected, this silences any incoming notes while the timeline is on this step. Note that the Mute Group must be enabled for the selected steps to be affected. The 'X' editor button on the far right clears all steps.

## Octave Shifts



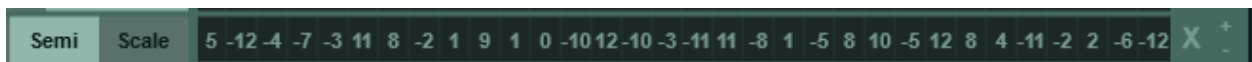
Each step has its own Octave shifting option. These allow you to shift any notes within a step up to 2 octaves up or down. The editors on the right are 'X' to clear all steps back to 0, '+' to increment all steps by 1, and '-' to decrement all steps by one.

## Passthroughs



Each step has its own Pass button, and when it is activated the incoming note will pass through the algorithm stages of the device unchanged – it will still be subject to scaling. Note that the Pass Group must be enabled for the selected steps to be affected. The 'X' button on the right is again for setting all values to the default.

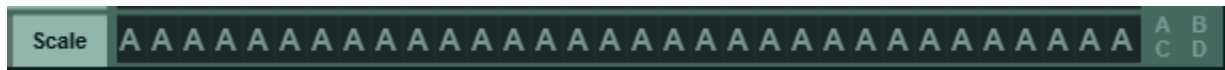
## Semitones



Very similar to the Octave shift, but for Semitone increments. Note that due to scaling, smaller changes to the values may not always have a noticeable effect on the output. The editors on the right are 'X' to clear all steps back to 0, '+' to increment all steps by 1, and '-' to decrement all steps by one.

## Scales (Sequencing)

Lastly, we have the Scale selection lane.



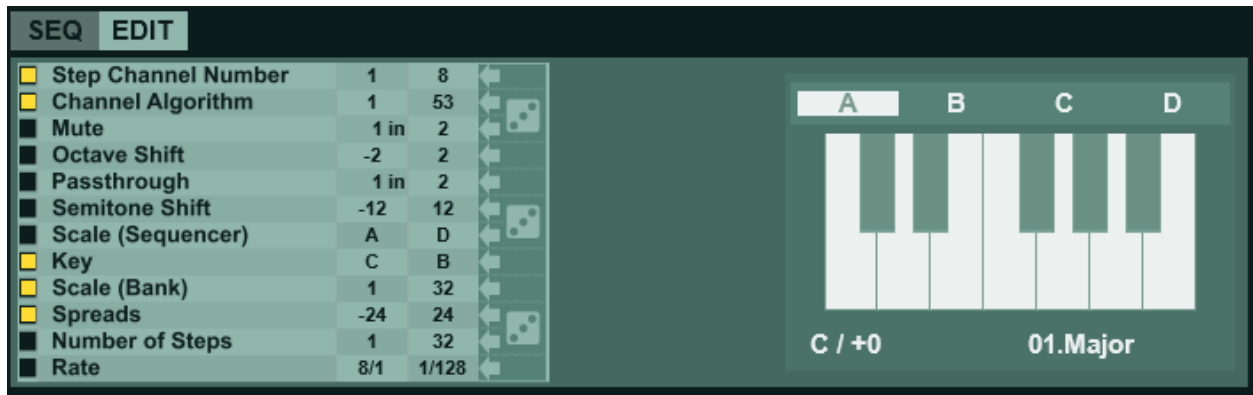
This lane is quite powerful on its own as you can use the Pass Mode to avoid algorithm usage and thus have a Sequencer which is purely for the Scales, giving a multi-scale capable sequencing tool. This is always defaulted to mode A as changing this without forethought would typically give dubious results.

## Spreads



There are 4 Spread controls in the Edit Panel. Each range from -24 to +24, with the A/B output types as explained above. Each value simply adds or subtracts the appropriate amount to the incoming notes, depending on which algorithm is selected.

# The Edit Panel

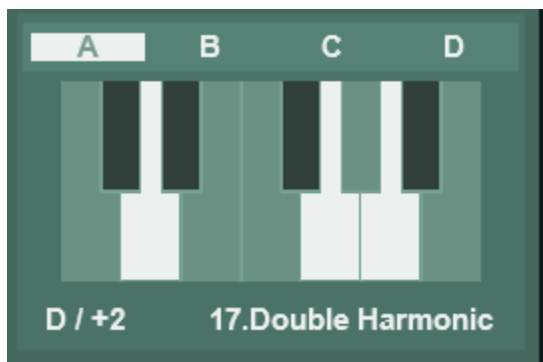


## Randomizer Settings

We wanted a really quick and easy way of setting up custom randomizers, and this seemed like the best way to do it. Simply use the toggle switches here to specify which values you want to be randomized. Then on the right hand side of the settings area you can specify the value ranges that the Randomize function will use.

Once you have the settings you want to randomize just hit the Randomize button on the Sequencer Panel, or alternatively you can use the dice roll area to right of the ranges, which allows you to randomize the individual areas.

## Scales



Finally, the Scales section sits near the output of the device, just before the Output Filter. There are several presets on board, and the keyboard interface is fully programmable.

## **The Included Patches**

We kept the patches to a minimum for this device. To put it simply, you won't know what works until you've fiddled around a bit, and every source of input material can produce varied results, so although it was necessary to include some "general ideas" patches to show different setups, this device is best used with patches as a cognitive aid rather than a replacement.



# **Using MeloDramatik with other Players**

All Player devices in the Propellerhead Shop are standards compliant with The Rack Extension SDK version 4.0 and onwards. This means that a certain set of behaviours are enforced, ensuring interoperability between all Player devices and how they receive and transmit data to other devices, including those which may comply with different standards.

As boring as the above paragraph may sound, it is hugely important when it comes to having fun musically using these devices. Effectively, it means that you are safe to assume that any Player is compatible with MeloDramatik. Good results are easily and safely obtained when using MeloDramatik alongside Note Echo, Scales and Chords, Dual Arp, and an ever-growing collection of third-party devices, including our own Loopcracker Suite devices, of course.

There are only two details which really need to be given consideration: Routing and Direct Record.

## **Routing**

Players will always work in a top to bottom order as presented on the rack screen. This starts at the main control unit at the top, which hosts the Direct Record switch and the Bypass All, along with the Send to Track button.

The routing of the Players is then in order from top to bottom unless something is done with CV if applicable connections are present on the device, or a special device is used to break the routing; there are currently 2 such devices in the Propellerhead Shop at the time of writing.

Finally, the note events reach the instrument at the bottom of the stack and **ONLY** that instrument. Should you have multiple instruments that you wish to send the Player notes to, then all the instruments will need to be present in a Combinator and the Players need to be external to that Combinator. Should you wish separate FX processing for each of these you then need to send the audio outputs to separate Mix Channels.

## **IMPORTANT**

If you wish to use our Note Humanizer plugin with MeloDramatik, it is crucial to understand that the routing will produce radically different results for each scenario. If the Humanizer is before MeloDramatik in the chain then the randomized micro differences in the timing will lead to different results every single time as the note order of chords or closely timed drum hits will change, and thus completely change how MeloDramatik's algorithm will calculate things. If you wish to get the "normal" MeloDramatik result and have it humanized, then you must insert Note Humanizer **AFTER** MeloDramatik.

## **Direct Record**

The Direct Record switch is arguably the most important standard feature of the Player stack. When it is engaged the Player will transmit the output results of whatever manipulations have been set up to the applicable sequencer track. This is all fairly simple if using a Player or CV device as the sole source of note generation, you just record straight into an empty instrument track.

If using data from the sequencer track itself though, another issue arises; a sequencer track cannot act as both source and target for Player notes, it is one or the other.

The solution to this is again to place the instruments in a Combinator, which means you then have two sequencer tracks to play with. From this

point you use one as the source and one as the target – the exact nature of this is dependent on what has been setup, and with what intent.

There are multiple possibilities, of course, but no seasoned user should struggle with it. If you are a novice then it's worth visiting one of the Reason user websites such as ReasonTalk and asking for advice, or one of the many social media platforms such as Facebook. We all start somewhere, and with the exception of one or two individuals who have social issues, you tend to find that the majority of the people in these groups are very friendly and wish to help.

Our own Facebook page will be fairly well monitored so if there is any specifics that you need covered then this is one of the best places to do it. Simply like the page, add yourself to the group and send us a message. Our response time will be "reasonable", and we will endeavour to give you the info you need.

# Acknowledgements

The following people made this possible, without them it wouldn't have happened...

## **All family and close friends**

Juhani Ekstrand and everybody at Reason Studios, Stockholm

Kenni Andruszkow and the rest of the fine folks at ReasonTalk

Navi Retlav and all at NaviRetlavSound

Giles Reaves and all at Selig Audio

Bes

Loque

Burningtoken

Diminished

Boombastix

MrFigg

Olive

Challism

All other testers and content providers have our massive gratitude.

Thanks to all the good folks on the various Reason-centric Facebook pages as well for giving moments to laugh, cry and feel inspired about.

Any questions? As in, reasonable ones not answered by this document?

We might not answer at all, if you send us a line to  
[enlightspeed@gmail.com](mailto:enlightspeed@gmail.com)



**ENLIGHTENSPEED**