MPCX MPCLIVE

BIBLE

BY ANDY AVGOUSTI (MPC-TUTOR)

The MPC X & MPC Live Bible

Written By Andy Avgousti

Book Edition: 1.3.0
This Edition Published by MPC-Samples.com, 2018

MPC-Samples.com is a trading name of Beat Box Digital LTD, a company registered in England & Wales, Company No. 6810062.

Table of Contents

000 How to Use this Book

Which MPCs are compatible with this book?

This (1.3) edition of ‘The MPC X & MPC Live Bible’ has been written to be compatible with the **Akai MPC Live** and **Akai MPC X** running firmware 2.2.1.

Hardware UI Vs Computer UI

Perhaps the most fundamental concept behind the MPC is its legendary ‘in the box’ workflow, so in this book I’ve set out to help you develop that unique and creative way of producing. To do this I present all tutorials from the perspective of the hardware controls and touchscreen, so whether you are working entirely in ‘standalone mode’ or if you are connected to your computer in ‘controller mode', I’ll teach you how to work entirely from the MPC hardware itself using the dials, buttons, pads and touchscreen for all functionality.

Recreating the Tutorial Examples

Every tutorial in this book comes complete with all the files you need to recreate the tutorials as you read through them. All these files can be found in the ‘**Tutorial Files**’ folder which was included in the ‘zip’ archive you downloaded after buying this ebook. To learn how to transfer this folder to your MPC, check out **Appendix B** at the end of this book.

Reading This Book On Mobile Devices

If you want to read the book on a more portable device such as a tablet, smartphone or ebook reader then I would suggest you use either the EPUB or MOBI version of the book found in the ‘**Ebook Reader Versions**’ folder (contained within the original zip file you downloaded). For more how to transfer an ebook to a mobile device, please refer to my article here: http://www.mpc-samples.com/article/transferring-mpc-ebooks-mobile-device

Initial MPC Set Up

If you are having trouble setting up your MPC, check out **Appendix A** at the back of the book which includes tips on general hardware configuration.

Release Notes For This Edition

If you would like to know what updates I have made to this book since the last edition, please go to **Appendix E** at the end of the book where you'll find an overview of the changes.

If you discover any problems or bugs in the book, please let me know via support@mpc-samples.com and I'll get them fixed as soon as possible.

So, fire up your MPC and let's get cracking!

001 Understanding The MPC Workflow

Traditionally the MPC has always been a very powerful combination of MIDI sequencer, audio sampler and controller – but in the MPC X and MPC Live, the MPC became a fully-fledged DAW. Let's take a closer look at what these terms all mean.

The MPC in a Nutshell

The MPC is a music making machine, some people call it a 'groovebox', others see it as a fully fledged 'DAW' (digital audio workstation).

From the very outset (back in the days of the original MPC60), 'MPC' stood for MIDI Production Center and at the heart of every 'MPC' to this very day is the **MIDI sequencer** which is capable of recording and playing back MIDI and audio performances over multiple tracks of information, allowing you to gradually build and arrange complex songs that utilise an unlimited source of sounds and instruments.

The Basic MPC Structure

When you begin to work on a new song idea, the starting point is always to begin with a new blank canvas called a **project**. A project will contain all the sounds, sequences, audio and settings for that entire song.

A project comprises of one or more **sequences** chained together to create complete **songs**. Each sequence consists of one or more **tracks,** with each track configured to record and play back a specific 'type' of data.

There are fundamentally two kinds of track in an MPC sequencer; a pure **audio track** and a **MIDI track**. An **audio track** holds a complete audio performance, often recorded from a live sound source or imported from a previously recorded digital file and will typically contain vocals, acoustic instruments, scratching and other 'live' performances.

A **MIDI track** lets us capture an MPC-based performance but not as 'audio' in the traditional sense – instead a sequencer records your performances as a series of electronic '**events'** (or 'instructions'). These instructions are recorded in a standardised electronic data format called **MIDI** (**M**usical **I**nstrument **D**igital **I**nterface).

The standard method of recording MIDI performances in an MPC is via the sixteen built in rubber pads. Each time you press a pad during your performance, a unique 'event' is created which captures every aspect of that particular pad press; the pad bank and number (e.g. A04), the velocity (how hard) it was hit, the duration of the pad hit (i.e. how long you held down the pad), the sequence time at which it was hit, as well as more obscure events as we'll see later in this book.

These 'instructions' can then be played back to re-trigger your pads exactly as they were originally played, thus re-generating the audio of your performance. A MIDI sequence performance can therefore be very easily edited at any time; in fact you can change every single aspect of each event, even changing what sound is triggered by that event.

MIDI can be used to trigger sounds from any sound source that understands MIDI data, such as piano, bass and synth sounds from externally connected sound modules and synths, or from software instrument plugins installed in your computer (such as VST instruments).

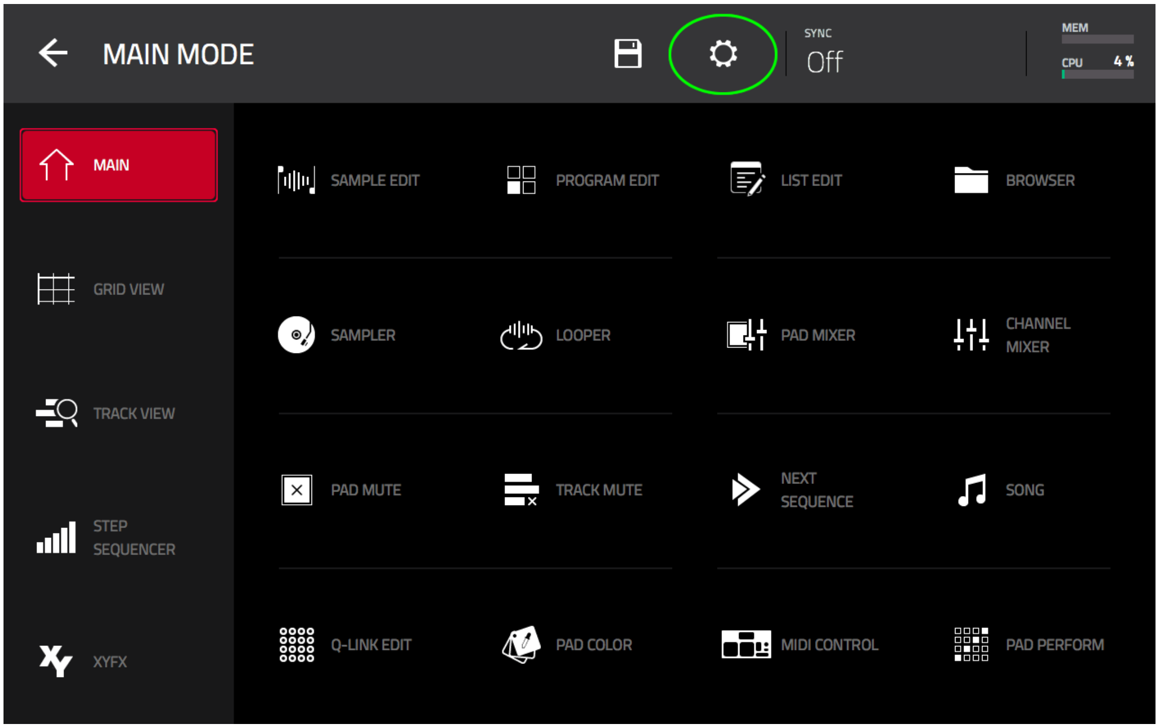
It's important to note that ***a MIDI sequence itself does not contain any audio whatsoever*** – it only plays back 'instructions' that will subsequently *trigger* audio from other devices, be it MPC sample programs, instrument plugins or MIDI sound modules and synths.

The MPC is actually a MIDI sound source itself as it features a built in audio **sampler** that allows you to load or record your own unique sounds and loops from vinyl, CDs, microphones etc. Once you have sampled or loaded sounds into your MPC, they can be manipulated using the various built in processing and editing tools and then assigned to a **program** which will allow you to apply further processing and effects to your sounds and also to play back these sounds using your MPC's 16 rubber pads – these pad performances can also be recorded and edited in your sequencer as MIDI data.

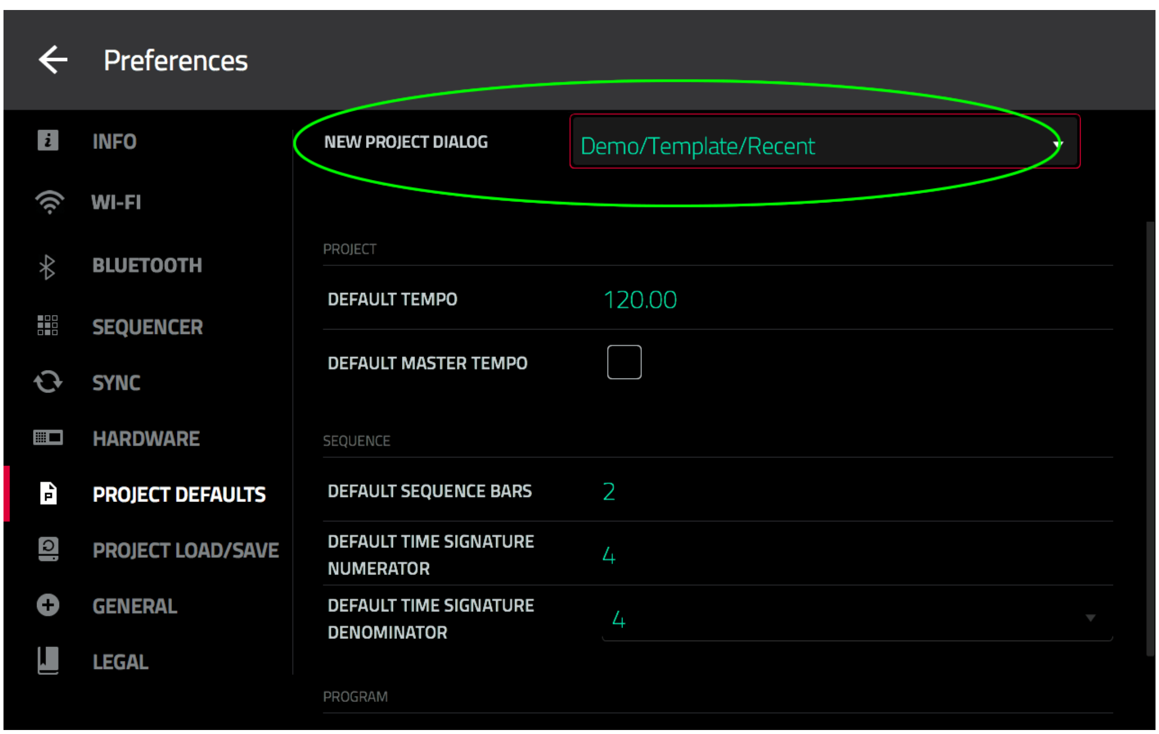
Sample programs are often used to play back drum sounds and drum kits, but can also be used to play back instruments, loops and sound fx.

Boot Up Options

Every time you switch on your MPC you'll be presented with the same initial screen unless you change the configuration in MPC '**Preferences**'. To set your preferred boot option, press the **MENU** button and then press the gear icon at the top of the screen:



This takes you to the **PREFERENCES** page; from the left side of the screen, select **PROJECT DEFAULTS**:



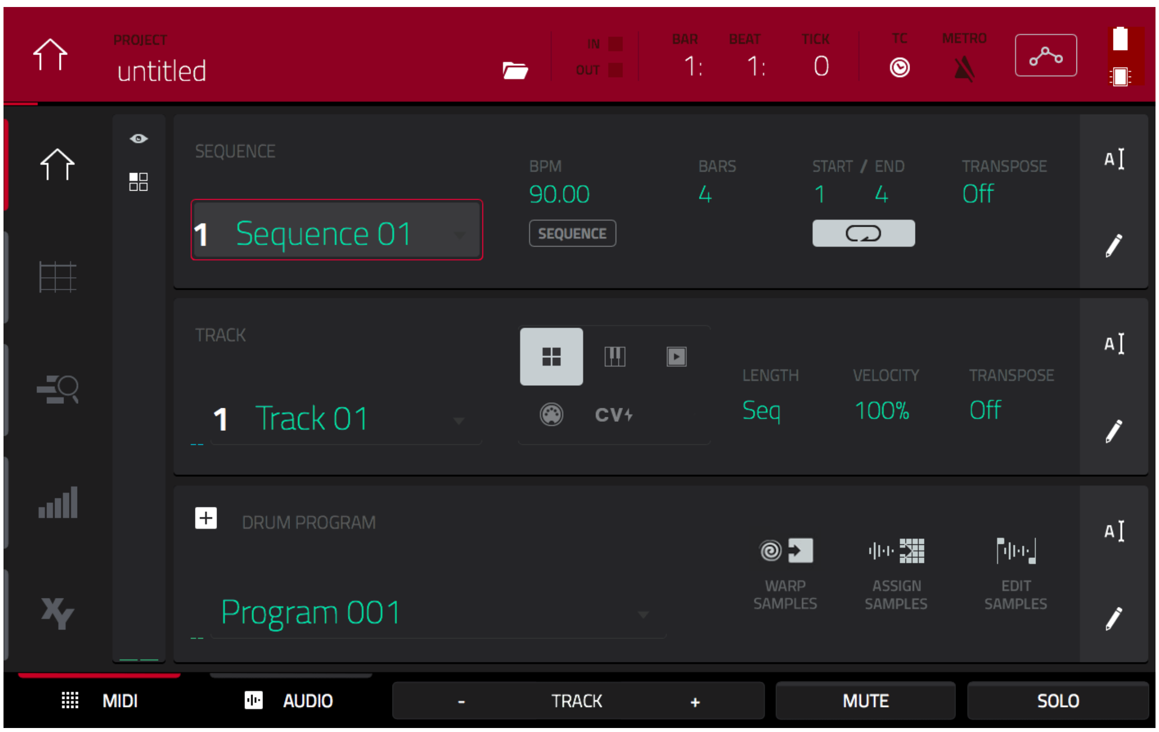
At the top of the page is the **NEW PROJECT DIALOG** option. There are 3 options to choose here; **Off**, **Demo** or **Demo/Template/Recent**.

For the purposes of this tutorial I initially suggest you set this to '**Off**' (I'll talk about the other two options in the next chapter). With this setting your MPC skips the initial 'landing page' and goes directly into a new blank project.

Your First Project – A Simple Drum Beat

With NEW PROJECT DIALOG set to 'Off', a fresh boot up of your MPC will take you directly into a fresh blank **project**; the first screen you see when you start with a new project is **MAIN** mode.

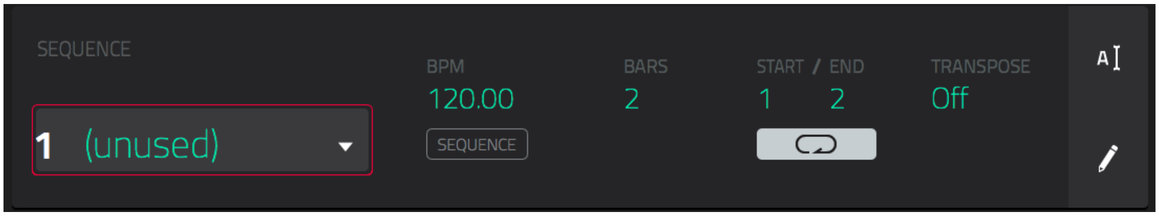
There are many different 'modes' in the MPC Software which provide an optimised environment for different tasks (e.g. 'Sample Edit' is for editing and processing samples, 'track view' is optimised for mixing tracks etc). MAIN mode serves as a 'hub', giving you a good overview of all your essential project variables.



As mentioned previously, at the heart of an MPC is its **sequencer** and MAIN mode is the place where you'll set up the first sequence for your new project. A project can contain as many sequences as your song requires (even just one very long sequence if you prefer!). Let's set up an MPC sequence so we can record a simple drum beat.

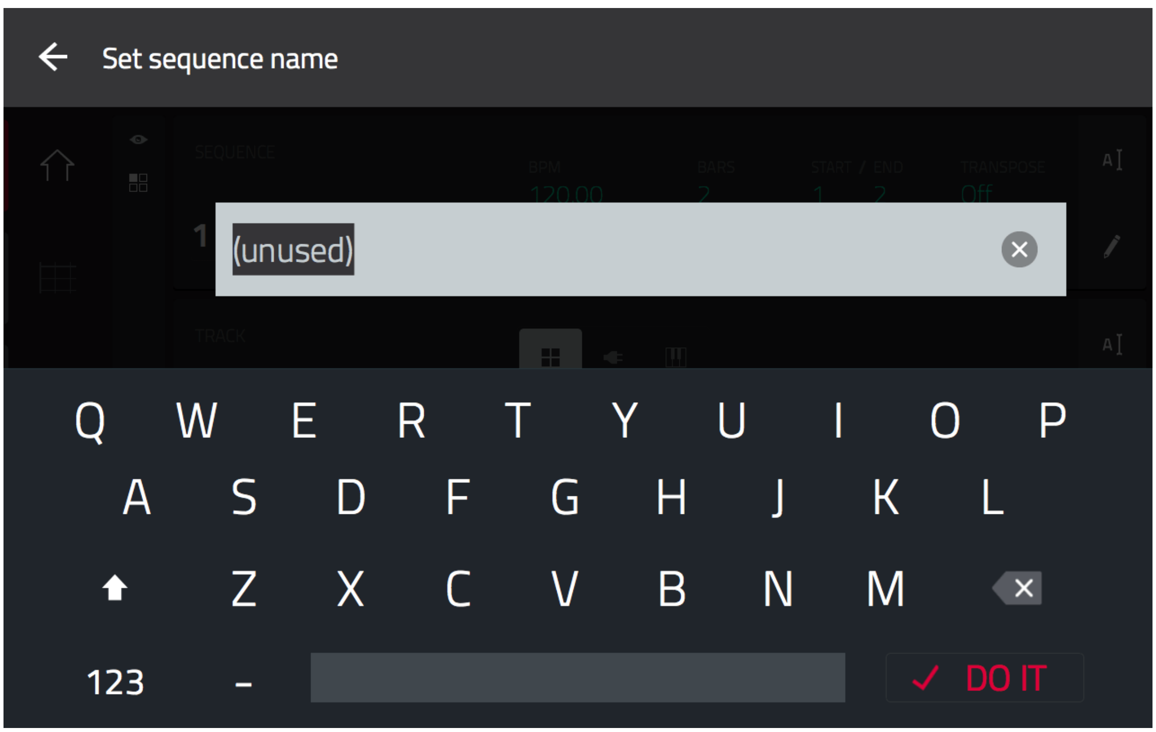
Setting up an MPC Sequence

The top row (**SEQUENCE**) has several parameters, so let's look at the most fundamental ones:

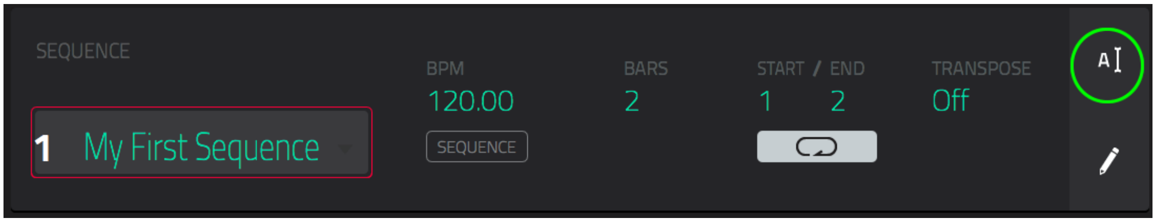


Sequence number & name

Each MPC project can contain up to 128 unique sequences, by default we start with sequence number **1** which is blank, as indicated by the '(**unused**)' label. To rename this default sequence, click on the '**A**' icon at the far right of the screen to bring up the '**Set Sequence Name'** screen.



Use the onscreen keyboard to give this sequence the name '**My First Sequence**' and hit **DO IT**.



A sequence also has a 'length', measured in '**BARS**', which is a standard musical measurement.

Leave this set to the default value of **2**.

Bars, Beats & Time Signatures

All music will have a time signature which dictates the way the music is played and structured. 99.9% of hip hop and dance music is played in a '4/4' time signature, which ultimately indicates that the music has 4 'beats' in each 'bar' (and each beat is played as a quarter note).

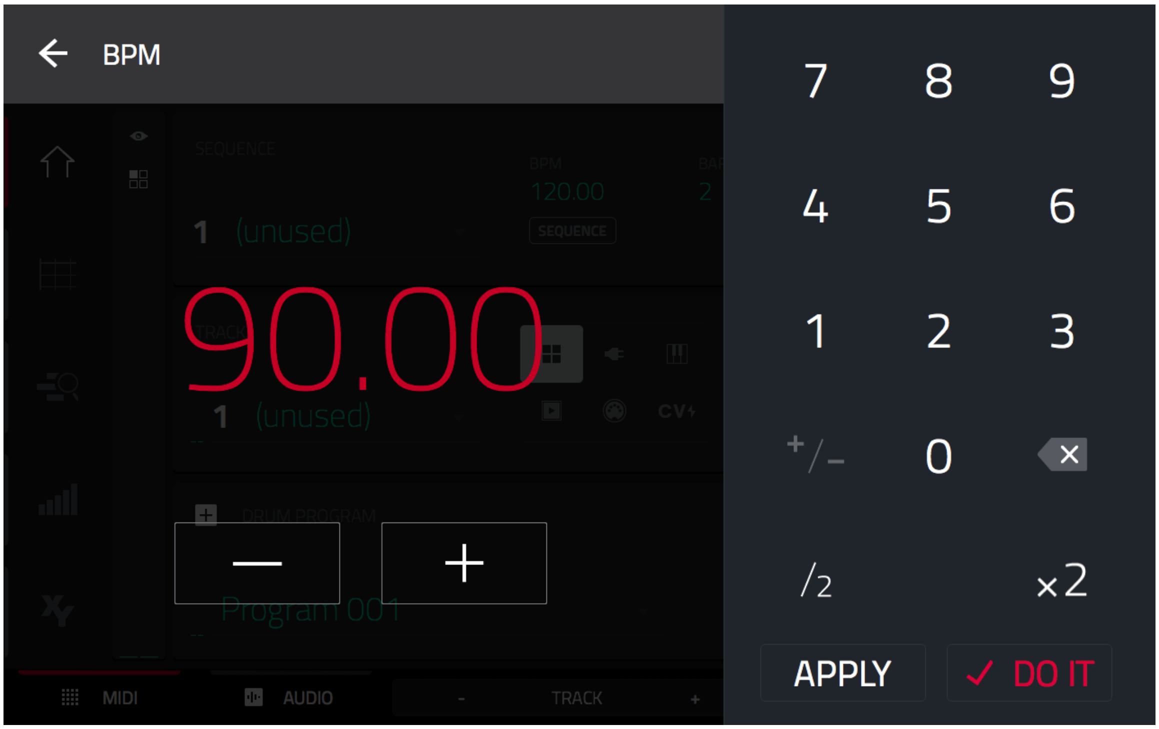
Simply put, a bar of 4/4 music is an equal count of '1-2-3-4', where each count is a beat. Two bars of 4/4 music is a count of 8, or if you prefer, two counts of '1-2-3-4'.

Sequence Tempo

A sequence also has a **tempo**, measured in 'beats per minute' (BPM) – this controls how fast your sequence will play.

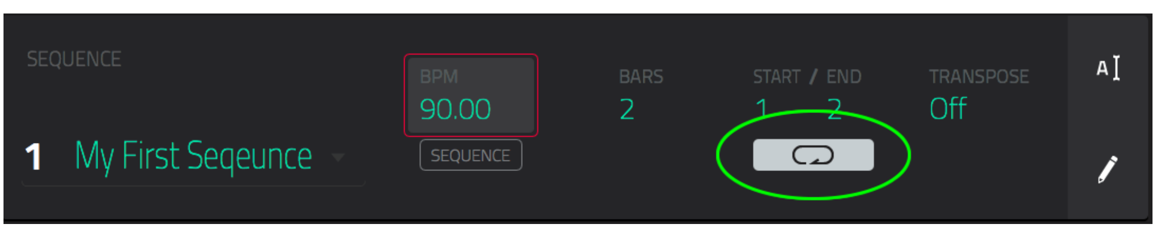
Let's change the BPM from the default 120.00 BPM to 90.0 BPM. Tap the **BPM** parameter with your finger to select and start turning the data wheel anti clockwise until the parameter reads **90.00**. Using the data wheel like this will decrease the BPM in single units – if you ever need more accuracy, hold down the **SHIFT** button while turning to move it 1/100th units.

Alternatively, you can double tap the BPM parameter to bring up the number pad:

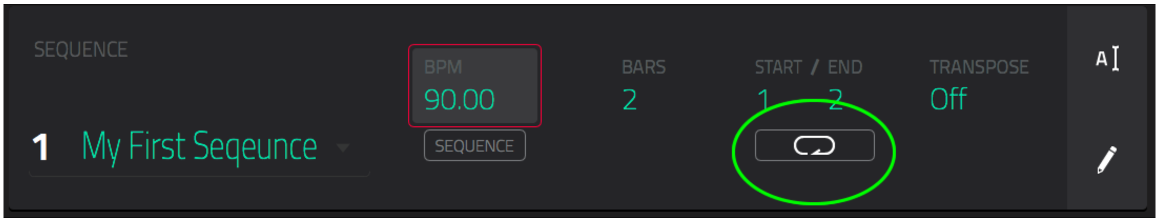


To set a tempo of 90.00 BPM, enter **9**, **0**, **0**, **0** and hit **DO IT**.

Sequence Loop

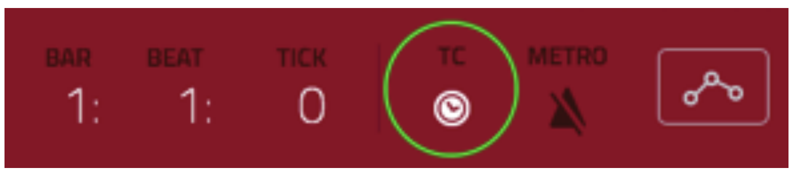


By default (with **LOOP** set to **ON**) when a sequence reaches its end (in this case, at the end of 2 bars) it will 'loop' back to the beginning of the sequence and carry on playing in an endless loop until you press the STOP button. Alternatively, it can simply carry on recording and just keep extending the length of the entire sequence accordingly until you press STOP (**LOOP** setting to '**OFF**') – for this first example, I want you to record for as long as you wish, so tap the loop setting to turn it **OFF**:

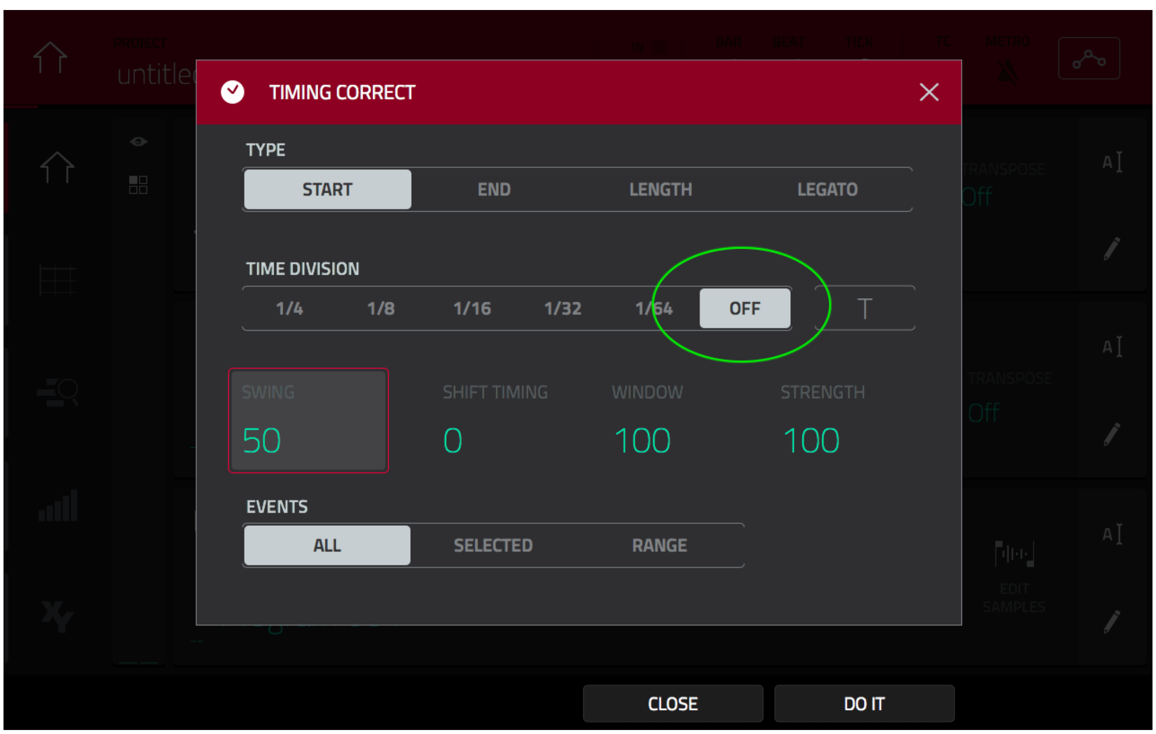


Timing Correct

'Timing Correct' (TC) is the MPC's method of trying to automatically fix sloppy playing! This is something I want to look at in more detail later in the book, so for the moment let's make sure this feature is turned off. To do this, press the **TC** icon at the top of the screen (it's the little clock icon):



This will bring up the **TIMING CORRECT** panel:

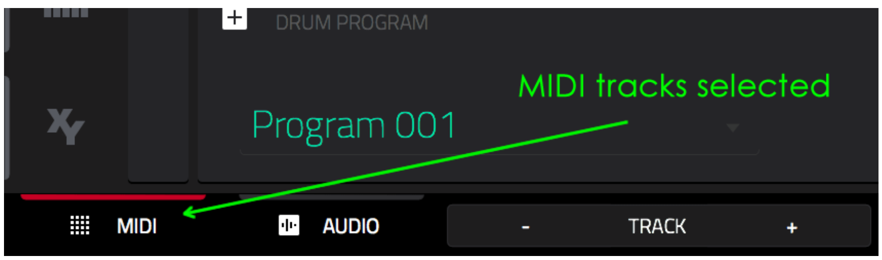


There's lots of settings here, but to turn off timing correct you just need to press **OFF** in the **TIME DIVISION** section – now press '**CLOSE**'.

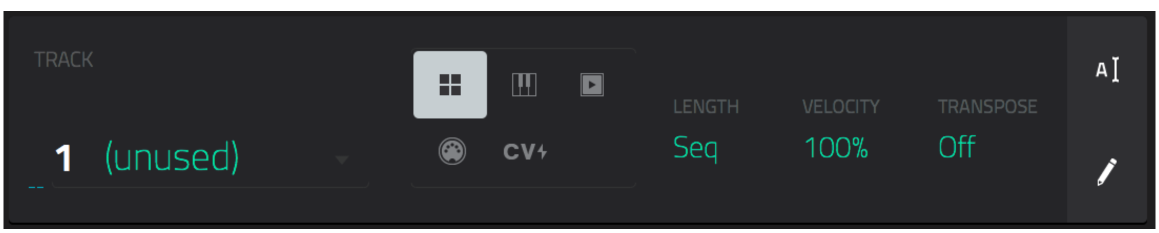
So let's quickly recap. We've started with a new blank project and have set up the first sequence ('**My First Sequence**') with a **tempo** of **90 BPM**. Our sequence is set to record everything exactly as we play it (**TC:OFF**) and as we've set the **sequence loop** to '**off**', it will record our entire performance until we press STOP.

Sequence 'Tracks'

Each sequence in an MPC must contain at least one 'track' to which we will record performance data. As mentioned previously, there are fundamentally two types of sequencer track in an MPC; an **audio track** or a **MIDI track**. The default display in MAIN is for handling MIDI tracks, as indicated by the **MIDI** tab/button at the bottom of the screen (notice the red line at the top):

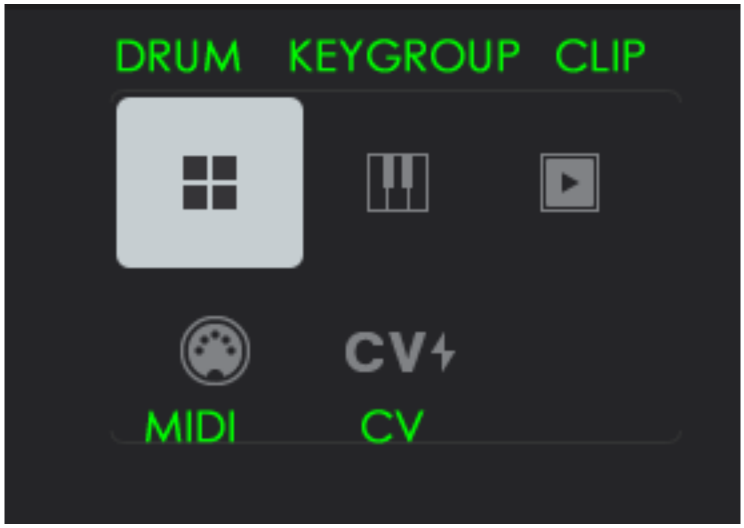


We'll look at audio tracks later in the book, for the moment let's stick with MIDI. Take a look at the '**Track Information**' panel:



This shows the currently *active* sequence track, which by default will be **track 1**. As this is currently blank it is shown as '**(unused)**'.

Each MIDI track in an MPC must be configured to communicate with a specific MIDI sound source; in standalone mode, the MPC currently recognises five different 'types' of MIDI device:



- **DRUM** – this refers to the internal ‘DRUM’ sample program format, which allows us to assign sounds to the MPC’s rubber pads (often drum sounds, hence the name).
- **KEYGROUP** – this is another internal sample program format, but this is designed specifically for playback of more complex sample-based instruments via your pads
- **CLIP** – this is the third type of internal sample program, optimised for triggering loops and phrases via the MPC pads.
- **MIDI –** use this type of track for triggering sounds from connected ‘hardware’ MIDI devices such as sound modules and synthesisers.
- **CV/GATE** – designed specifically for triggering sounds from an analog synth via CV/Gate outputs (MPC X only).

Additionally, while in ‘controller mode’ (when your MPC is controlling the MPC Software application on your computer), there is a sixth program type; **PLUGIN –** this is used to trigger sounds from any installed software instrument (including third party VST instruments).

Let’s leave the track set to **DRUM**. This means we are going to record a performance to this track using a standard MPC ‘DRUM’ sample program.

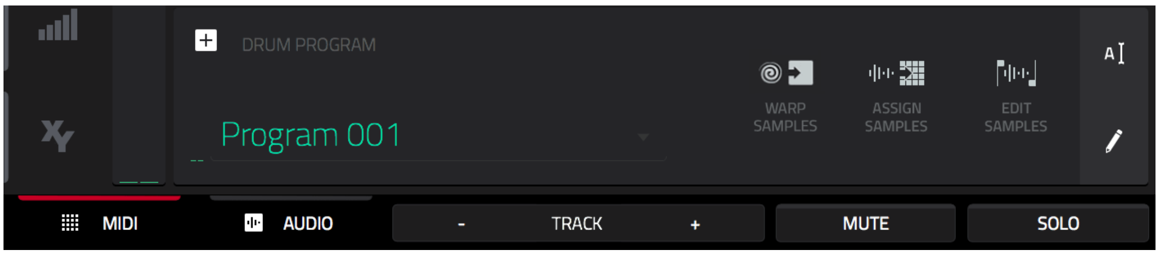
Understanding DRUM programs

At its most basic level, a DRUM program allows us to assign sounds to the rubber pads in the MPC hardware. Once a sound is assigned to a pad, each time you hit that pad you should hear that sound play back.

In addition to assigning specific sounds to specific pads, a DRUM program also allows us to perform a number of different sonic manipulations to each pad which change the way the sound on that pad plays back. Your project can contain any number of DRUM programs, each one containing its own set of samples and pad parameter configurations.

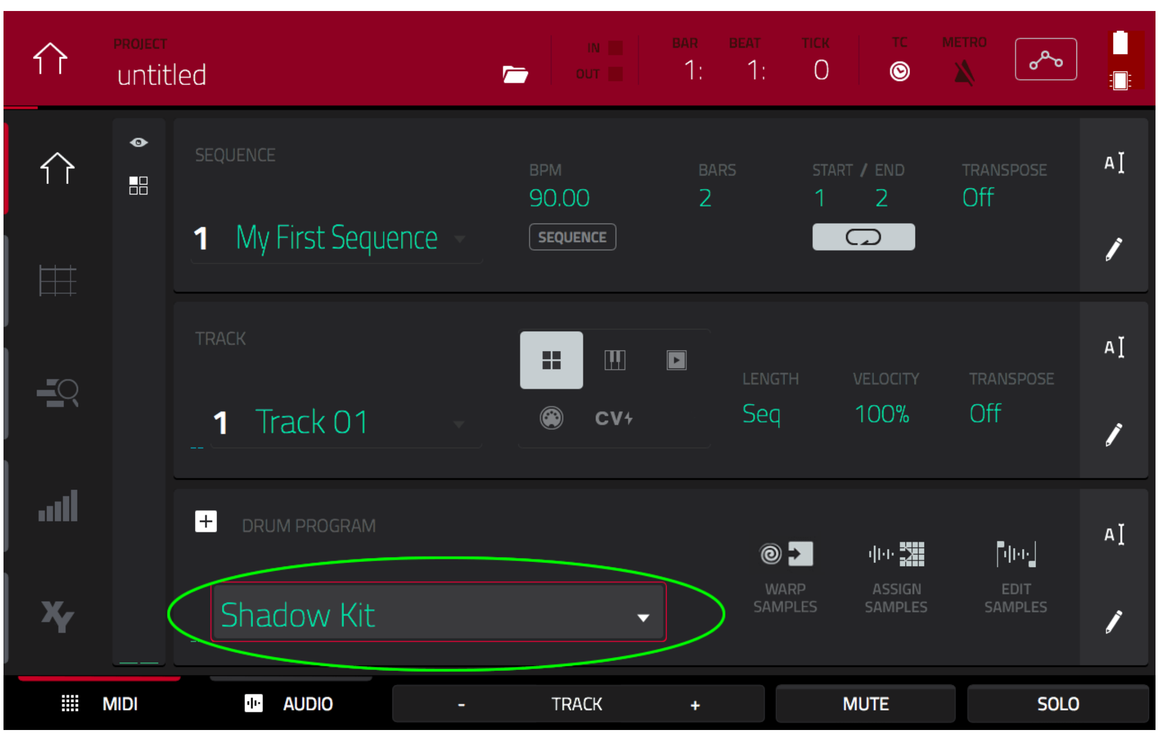
While the name ‘DRUM program’ may suggest that they are only suitable for ‘drum’ samples, this is definitely not the case, you can assign any type of sound to a pad – drums, percussion, bass, piano, vocals, hits, sound effects, loops and so on. However, when it comes to building drum kits, the DRUM program is the perfect choice.

When you start a new project in your MPC it creates a single blank DRUM program automatically – check out the third row of data in **MAIN**:

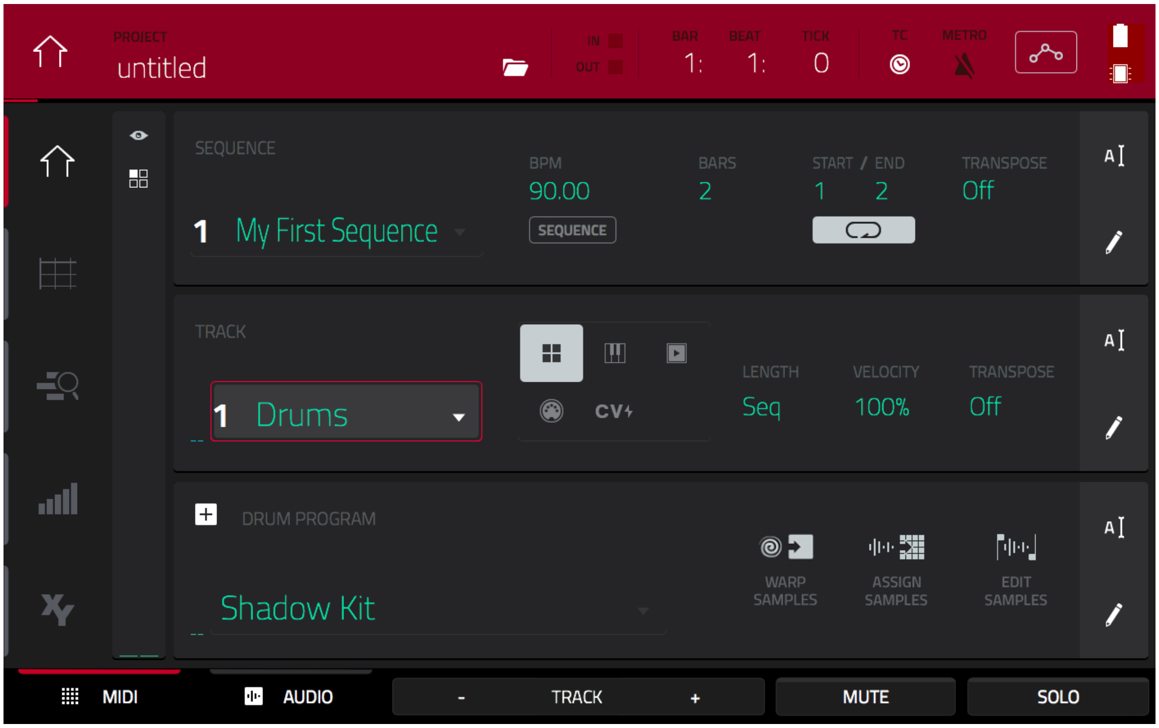


This row confirms that we have assigned a **DRUM PROGRAM** to the currently selected sequencer track (track 1). This DRUM program is called **Program 001** (which is the default name for any newly created sample program).

If you click on the ‘**A**’ icon at the end of this **DRUM PROGRAM** row it will bring up the ‘**Set Program Name’** screen – rename Program 001 to ‘**Shadow Kit**’.



It's a good idea to get into the habit of giving everything a name rather than using the defaults. As we are going to record drums to this sequencer track, tap the **'A'** icon at the end of the **TRACK** panel and rename this track to '**Drums**'. Hit **DO IT**.



Standalone Vs Controller Mode

Your MPC is capable of operating in two distinct 'modes'; **standalone** and **controller** mode. In '**standalone**' mode, your MPC runs from its own internal installation of the MPC Software using the CPU and memory available inside your MPC itself. In **controller** mode your MPC is connected via USB to your computer and 'controls' the MPC Software application installed inside your computer. Here the 'MPC' uses the CPU and memory of your computer.

If you are using the touchscreen interface, for the majority of actions there is no noticeable difference between running in standalone or controller mode. Throughout this book I tend to assume you will be running your MPC in 'standalone mode' although this is not a requirement as all the tutorials will also work the same in controller mode. There are however a couple of tutorials where I show you an advantage of working in controller mode (for example, so you can use third party plugins).

002 Building a Simple Drum Kit

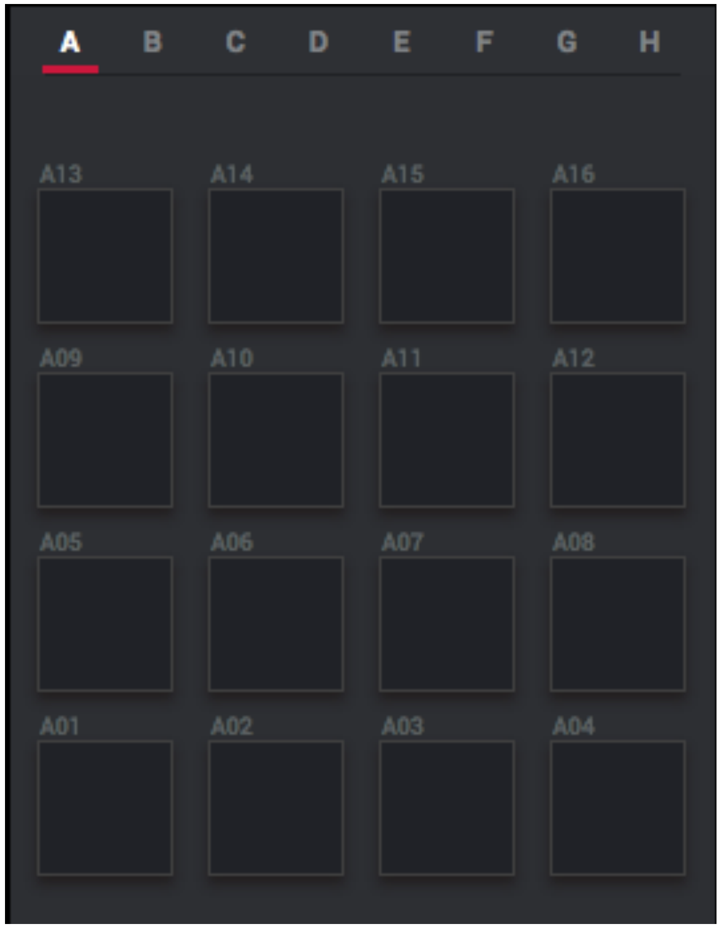
Before we can record a drum beat, we'll need to assign some drum samples to the pads in our blank 'Shadow Kit' DRUM program. Let's build our first drum kit!

Before going any further in this tutorial, please make sure that the '**Tutorial Files**' folder that came with this book has been copied to a removable disk attached to your MPC. More information on how to do this is included in **Appendix A** at the end of this book.

DRUM programs & Pads

Each DRUM program in your MPC comprises of 128 pads - we can assign samples to each of these pads so the pad will play back those sounds when the pads are hit. Now, your MPC hardware has only 16 *physical* pads, so to access these other 112 pads, we have the concept of **pad banks**.

The default pad bank selected upon boot up is **BANK A.** While you are in BANK A, the bottom left pad on your MPC is referred to as pad A01, while the pad at the top right is pad A16.

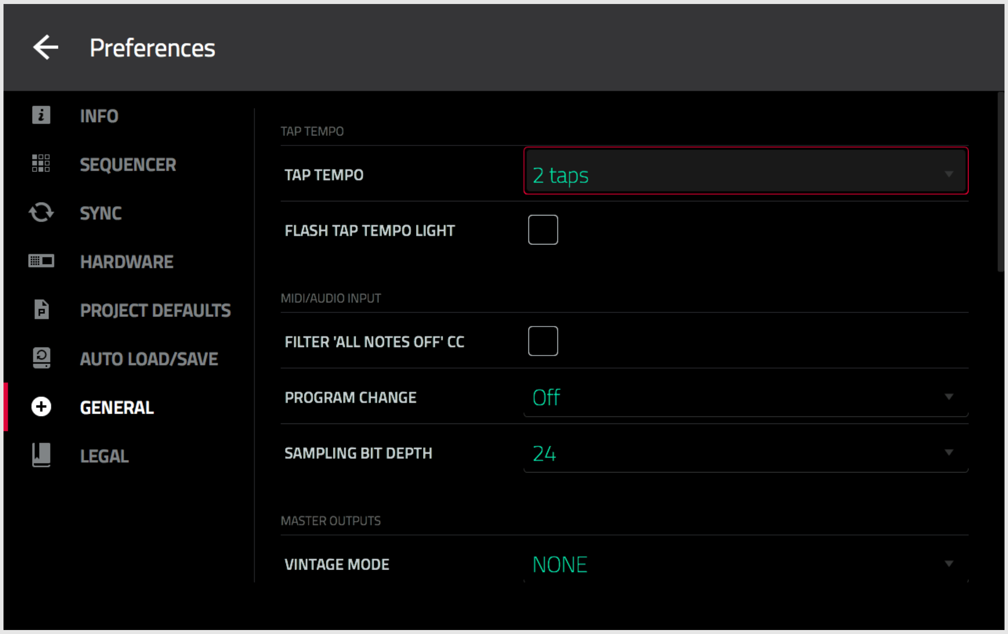


To access the next 16 pads in the DRUM program (pads 17 to 32), we press the **BANK B** button on the MPC hardware. Now the bottom left pad is **B01**, while the top right pad is **B16**.

When we get to **BANK E**, you'll notice there isn't a dedicated BANK E button – to access **BANK E**, simply 'double tap' **BANK A** – the button will illuminate in yellow instead of pink, indicating it is now in the BANK indicated in the writing underneath the button (E). Alternatively, instead of double tapping, hold down the **SHIFT** button and press BANK A. Do the same with buttons B to D to access banks F to H respectively.

Changing Pad Bank Behaviour

If you don't like the double tapping or shift tap options, there is a third option at your disposal. Go to **MENU > SETTINGS** (the 'gear' icon, top centre of screen) – in the **PREFERENCES** page, select **GENERAL**:



Double tap **BANK BUTTON PRESS** to give you two options; the first is **Select A-D**. This is the default action we've already described. Now instead choose '**Select/Toggle Bank**' and press **BANK A** a few times, slowly. This time the BANK A button toggles between A and E. I personally prefer this option.

While not a requirement, the standard convention when building a drum kit is to begin assigning sounds in BANK A and start using other banks only once you run out of available pads in BANK A.

Before we proceed, make sure **BANK A** is the currently selected BANK in your MPC.

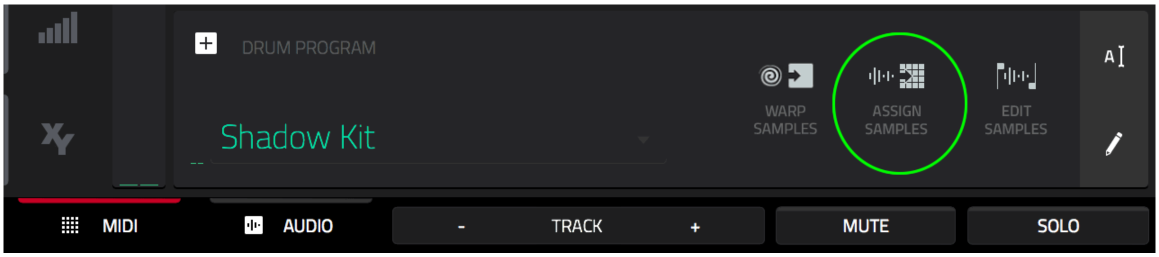
Loading & Assigning Sounds

Return to **MAIN**. To assign some drum samples to our kit we'll first have to source some drum samples!

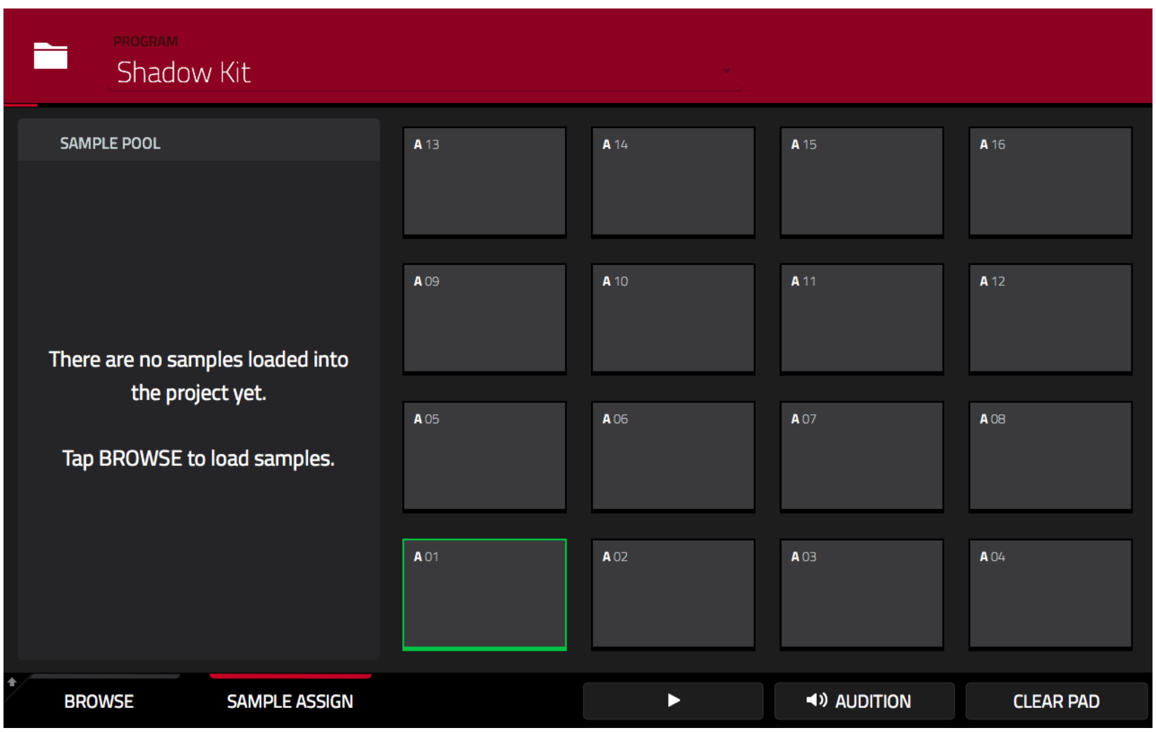
1. Load them as digital audio files from disk
2. 'Record' them from an external audio source

The process of recording the audio into the MPC (referred to as 'sampling') will be covered in a later tutorial later in this book, so at this stage we'll concentrate on the loading option.

On the **DRUM PROGRAM** row, make sure the Shadow Kit is the currently assigned program and tap the **ASSIGN SAMPLES** button at the far right of the row:



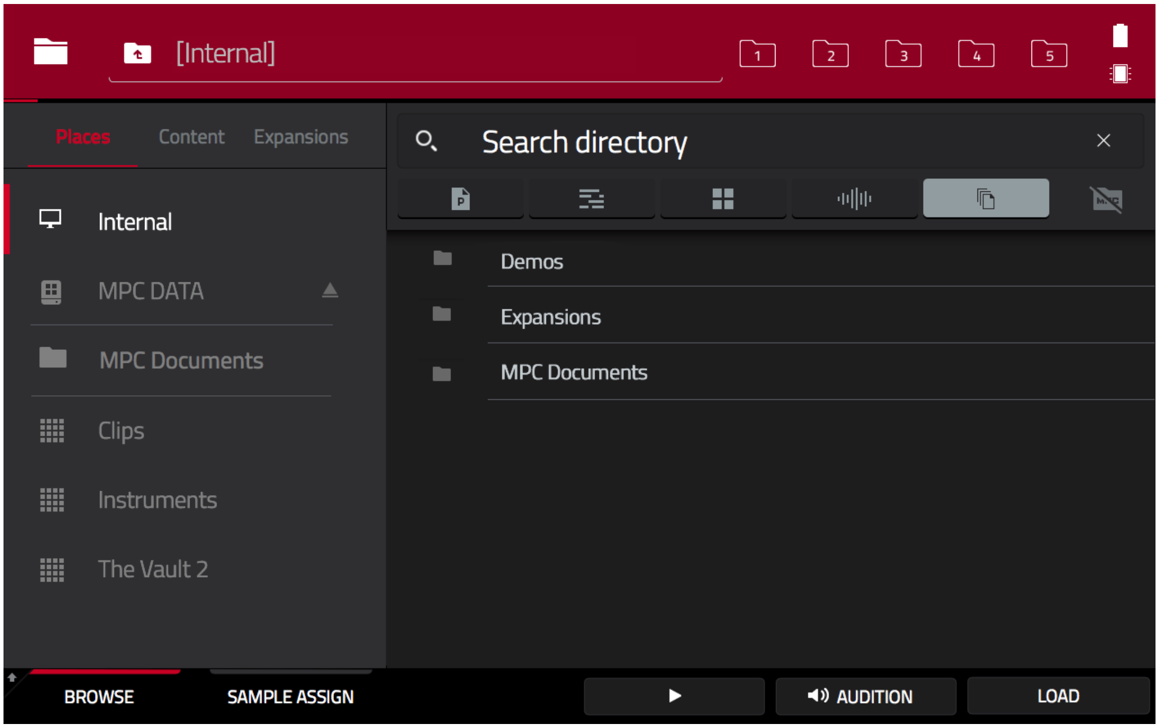
This will take you to the **SAMPLE ASSIGN** screen:



Here you can assign a sound to any pad in your current program. Hit **pad A01** – you'll see it becomes green, both on screen and the physical pad itself which indicates that this is the pad we're about to assign a sample to.

The sounds you can assign are held in the **SAMPLE POOL** at the left side of the screen. As you can see we currently do not have any sounds loaded into our project, so let's head over to the **BROWSER** and change all that.

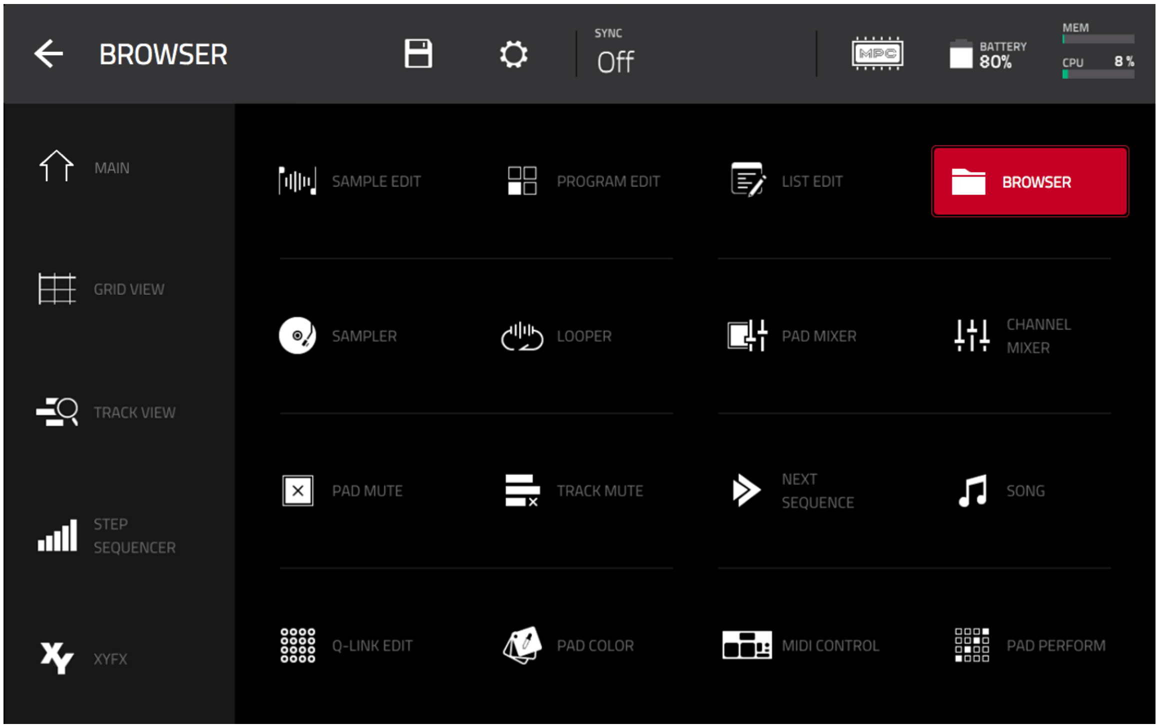
Hit the **BROWSE** button at the left bottom of the screen. This will take you directly to the MPC's file **BROWSER**:



The **File Browser** is a dedicated, bespoke file loading system designed specifically for the MPC. It can load any compatible file type into your project including sounds (in WAV, AIFF, SND and MP3 format), programs, sequences, and entire project files.

It's often convenient to access the BROWSER from the ASSIGN SAMPLES option from MAIN (as we just did), otherwise you can quickly get to the BROWSER by pressing the dedicated **BROWSE** button (MPC X), or by quickly double tapping the **MENU** button (MPC Live).

Another alternative is to go via the **MENU** screen – single tap the **MENU** button:

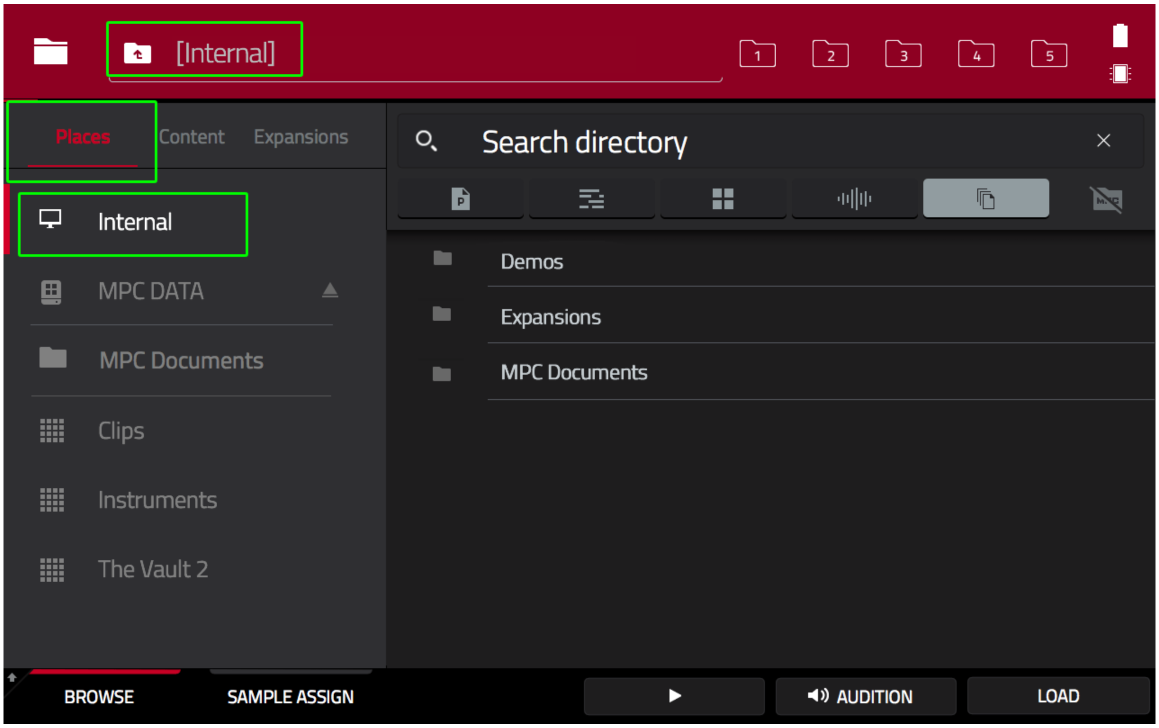


MENU is the 'central hub' where you can access all the various editing and performing modes of the MPC. Live owners will use the MENU a fair bit because, unlike the MPC X, you don't have dedicated buttons for most of the editing modes.

To go to the **BROWSER** from the **MENU** screen, simply single tap the '**BROWSER'** icon at the top right of the screen.

Navigating The File Browser

After launching the **BROWSER**, tap on the red **Places** tab near the top left of your touchscreen. On the left hand side you'll see a list of shortcuts to various different disk locations available to your MPC. Tap on the entry for '**Internal**':



‘**Internal**’ refers to the internal 16GB drive inside your MPC. When you tap on an entry for any disk location, the right hand side of the screen shows its contents. Here you can see that the **Internal** drive contains the pre-installed demo projects (‘Demos’), a folder for the included **Expansions** content (e.g. Vault 2) and an ‘**MPC Documents**’ folder.

Underneath ‘**Internal**’ you’ll see entries for any additional disks you may have attached to your MPC. This can include an SD card and any USB drives attached to the ports at the back of your MPC (you can see I have attached a disk called ‘MPC DATA’). And if you have installed an additional internal SATA drive, this will appear here also.

Which Disk Format?

Most disks you purchase will already be formatted; your MPC should be able to read most disk formats but for removable drives I’d recommend **exFAT** format for optimal results (compared to the older FAT32). If you install an internal SATA disk you could also use NTFS (HFS+ is read-only). Remember the disk format you choose should be compatible with your computer to allow easy file transfer. If in doubt exFAT is a good choice.

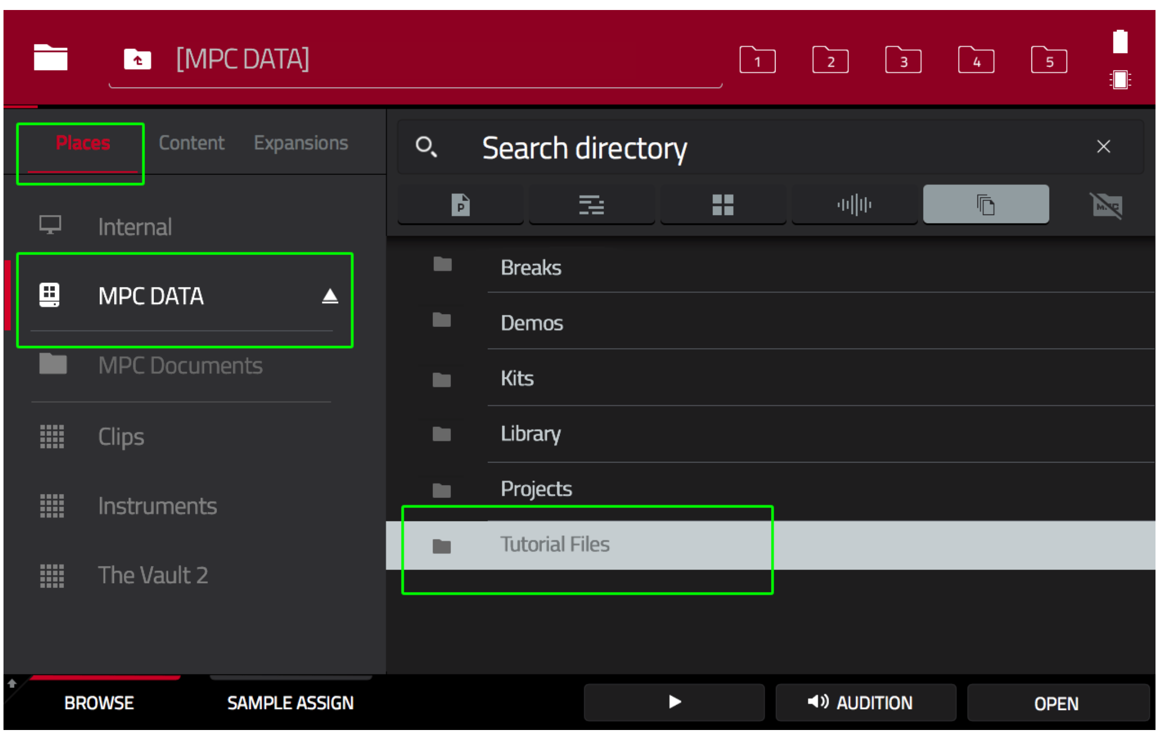
You can re-format a previously formatted disk within the MPC itself; while in the **BROWSER** screen, select the disk you wish to format (from the **Places** tab), hold down **SHIFT** and select **FORMAT DRIVE**. This will format the disk in exFAT format. Remember formatting a disk will wipe all existing data!

If your disk is not formatted at all the MPC cannot format it (or even recognise it) so you will need to first format it in your computer.

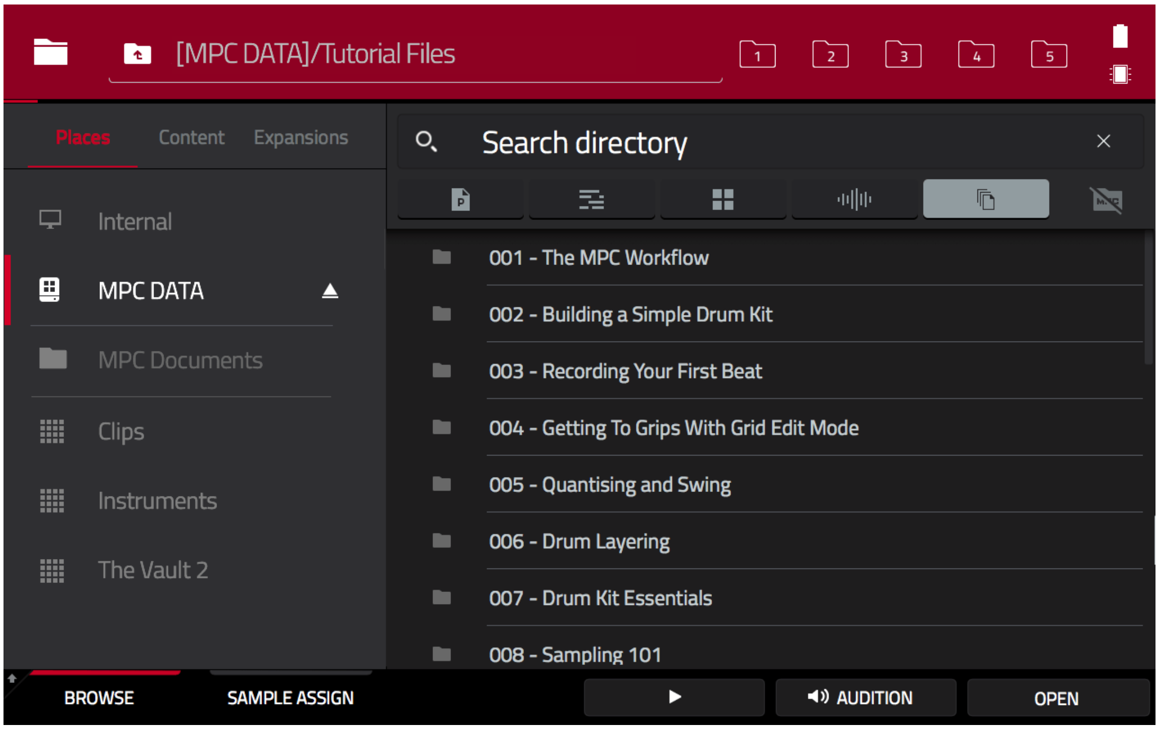
Loading sounds Into a Project

Let's navigate to the '**Tutorial Files**' folder so we can load some sample content into our project – this is the folder I included in the original 'zip' archive you downloaded after purchasing the book which contains all the files you'll need to recreate all the step-by-step projects in this book. I'm assuming that this has been copied to an external drive, such as a USB flash drive or SD card. If you need step-by-step guidance on how to do this, please refer to **Appendix B** at the end of this book.

Under the **Places** column, locate the entry for the removable disk containing the 'Tutorial Files' folder – in my example the disk is called '**MPC DATA**'. Tap on **MPC DATA** to enter that disk. At this point you'll see the contents of the MPC DATA disk appear on the right hand side of the screen (the MPC will only display files that it is able to read, e.g. sounds, MPC programs, MIDI sequences etc – but it will also display folders regardless of their actual content).



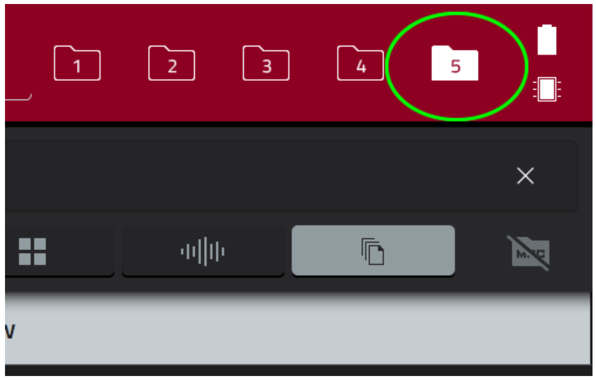
Scroll the data wheel or drag the files/folder list with your finger until you locate the '**Tutorial Files**' folder that you copied there previously (you can also use the or use the +/- buttons). Enter the folder by double tapping its name, or single tap to select it and then tap the '**OPEN'** button at the bottom right of the touchscreen - you are now inside the '**Tutorial Files**' folder, which consists of a number of sub folders for each chapter of the book:



Now, if you think you'll be wanting to repeatedly visit a particular location on your system, it's probably a good idea to 'bookmark' this location, and luckily Akai have provided five user-definable shortcut links.

To the right of the Folder Path you'll see the '**Folder Shortcuts'**. Pressing these will take you to a location on your computer that you've previously defined. By default these shortcuts are initially pre-set for you, but all these can be overridden with your own preferred location.

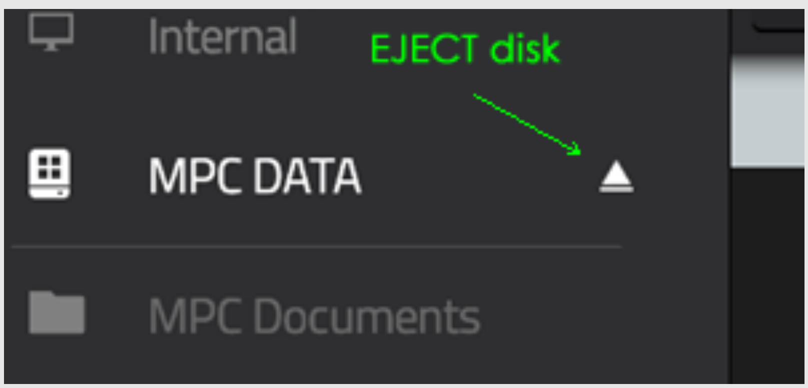
Let's replace **shortcut 5** with a shortcut to our '**Tutorial Files**' folder. To do this, make sure you are definitely *inside* the Tutorial Files folder, hold down the MPC's **SHIFT** button and while it is still held down, tap the '**5**' folder at the top right of your screen. You should see it turn white:



Now at any point you wish to navigate to the 'Tutorial Files' folder, you can instantly access it by touching folder shortcut '5'. You can perform similar steps to set up shortcuts on folders 1-4.

Safely Removing Disks From your MPC

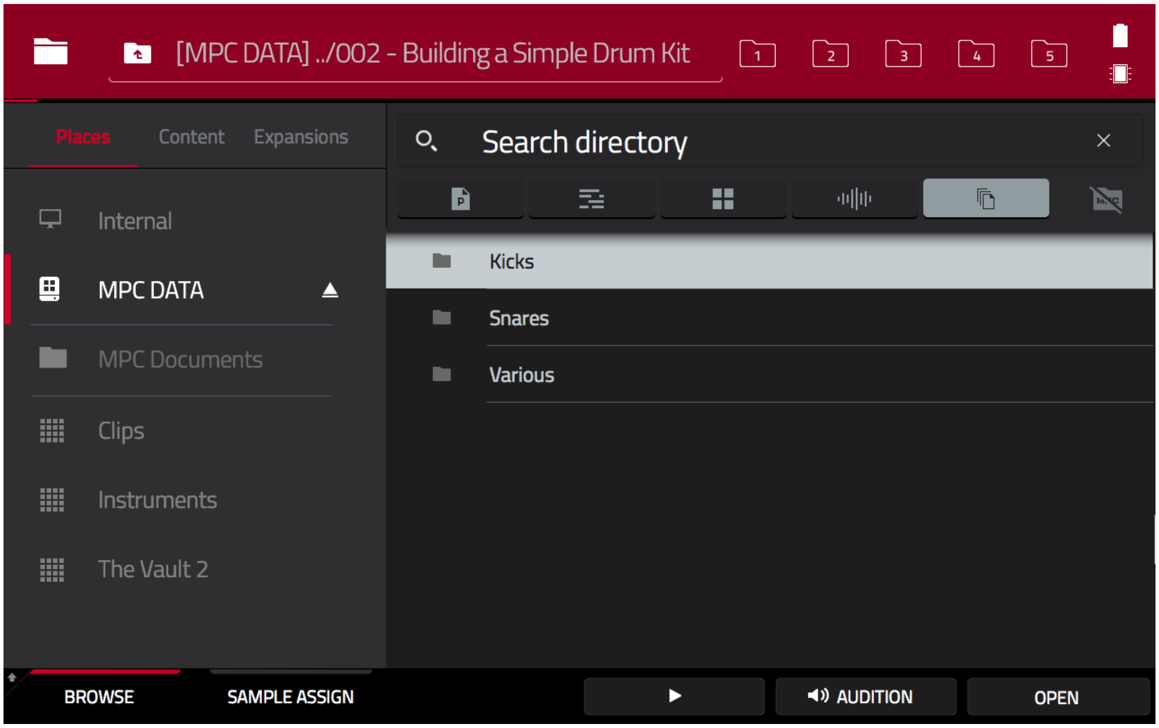
Next to each USB drive or SD card entry in the BROWSER you'll see an **eject** icon:



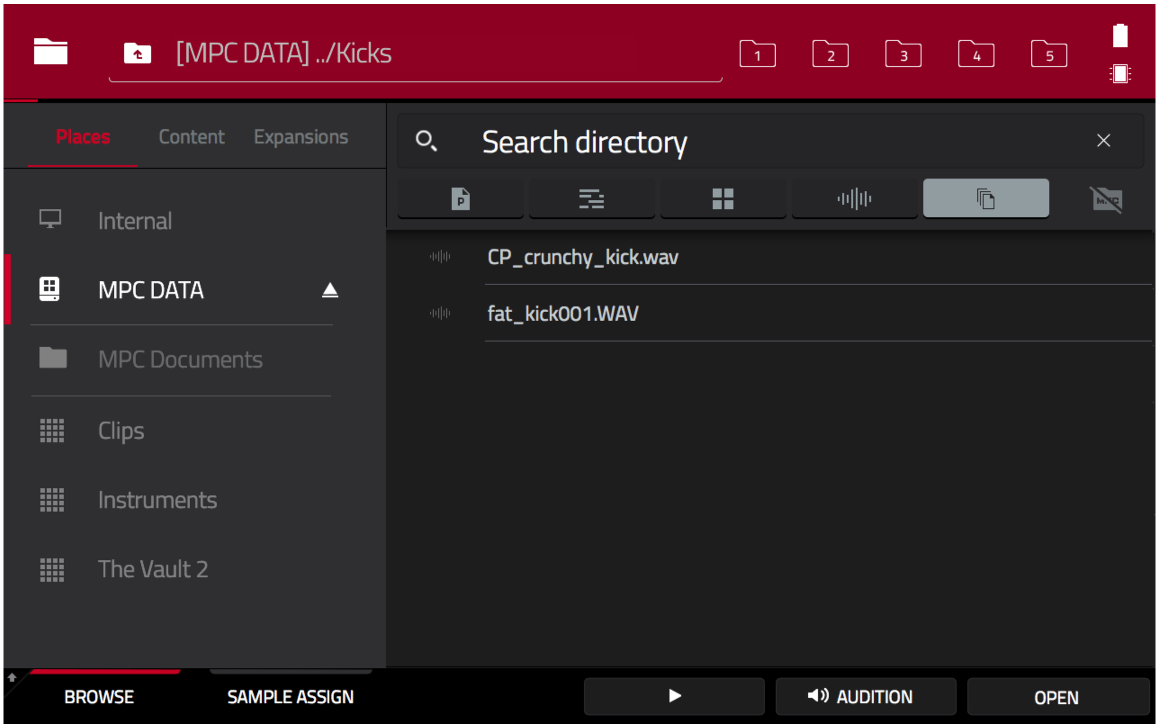
Before removing this disk from your MPC, remember to press the eject icon and select EJECT, otherwise it's possible you could damage the disk. The same is true if you connect your MPC to your computer via 'controller mode'; always use your computer's 'eject' option before disconnecting your MPC.

Loading & Assigning Sounds Via the Browser

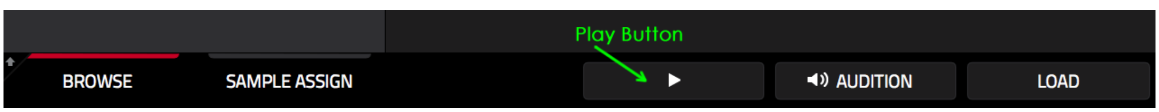
If you are not in the '**Tutorial Files**' folder, go straight to it by pressing the **Folder 5** shortcut. From the folder list, double tap the **'002 – Building a Simple Drum Kit**' folder to enter it:



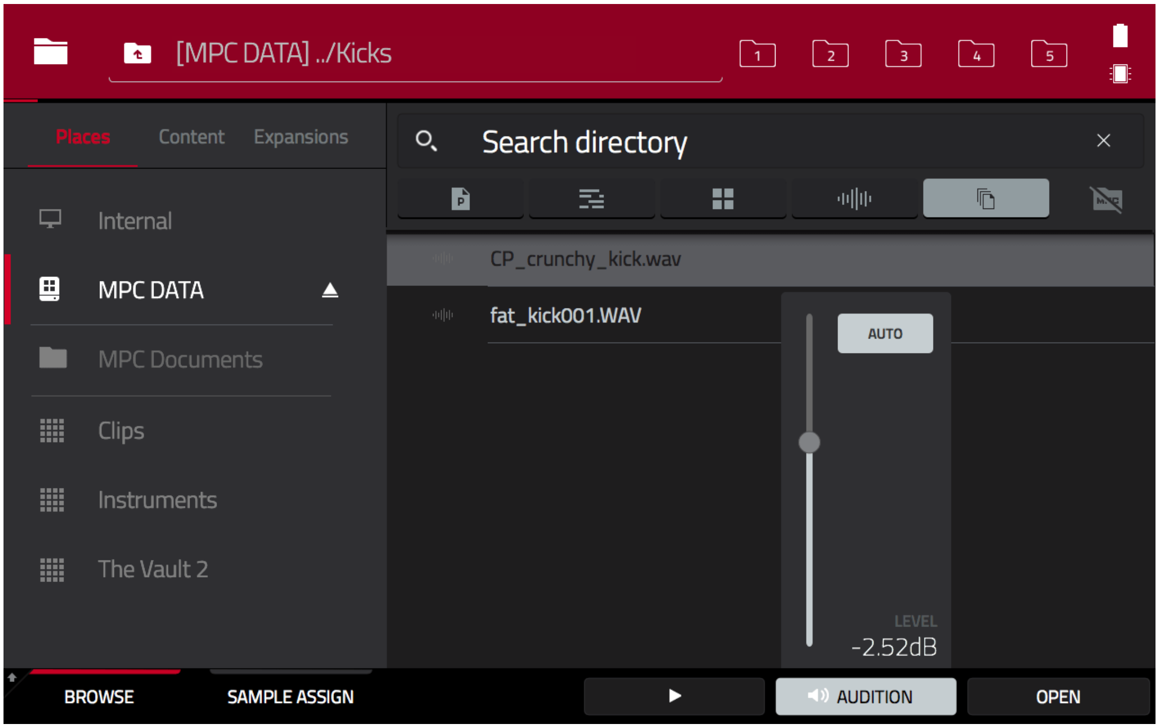
Here you can see a list of sub folders inside; double tap the '**Kicks**' folder:



Inside you'll find two kick drum samples. To preview any sound in the list you can single tap it and press the '**Play**' button at the bottom of the screen:



Alternatively, you can set samples to **AUTO PLAY**. To do this, first press the **AUDITION** button:

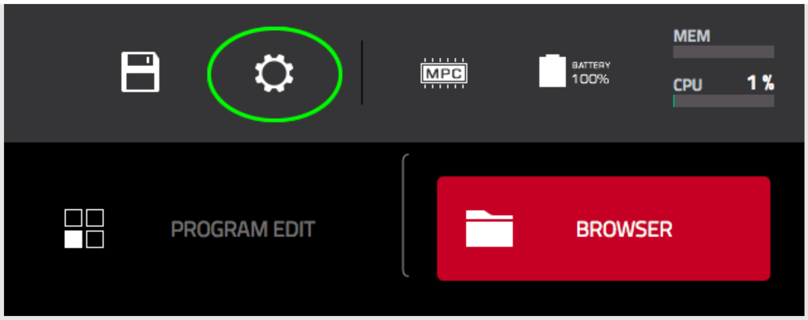


Press the '**AUTO**' button at the top of the pop up so it is selected as shown above. Now each time a sample is selected, it will automatically play back for you. You can also adjust the volume of the audition by dragging the provided slider up or down. The AUDITION pop up remains visible until you press the AUDITION button again.

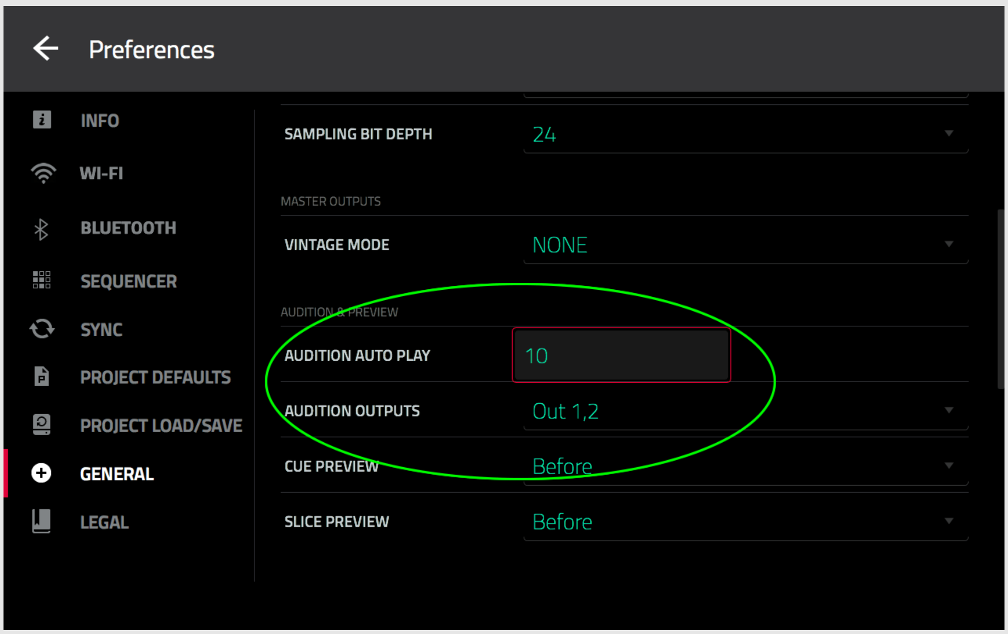
You can also stop a preview simply by selecting a new sound in the Browser; this stops the previous sample playing and plays the new sound instead. You can also manually stop an audition by pressing the **STOP** transport button twice on your MPC hardware.

Controlling Audition Length

If you are previewing entire songs in a folder, you might want to limit the length of the preview. You can configure this on the **Preferences** page. Go to **MENU** and tap the **Preferences 'gear'** icon at the top right of the screen:



Now go to **GENERAL > AUDITION AUTO PLAY** (you'll need to scroll down to see this):



To select a different audition length, single tap the default '10' and turn your data wheel anti clockwise. Set it to '**0**' for unlimited playback length (the sample will play in its entirety) or set it to a number from 1 to 10 (seconds).

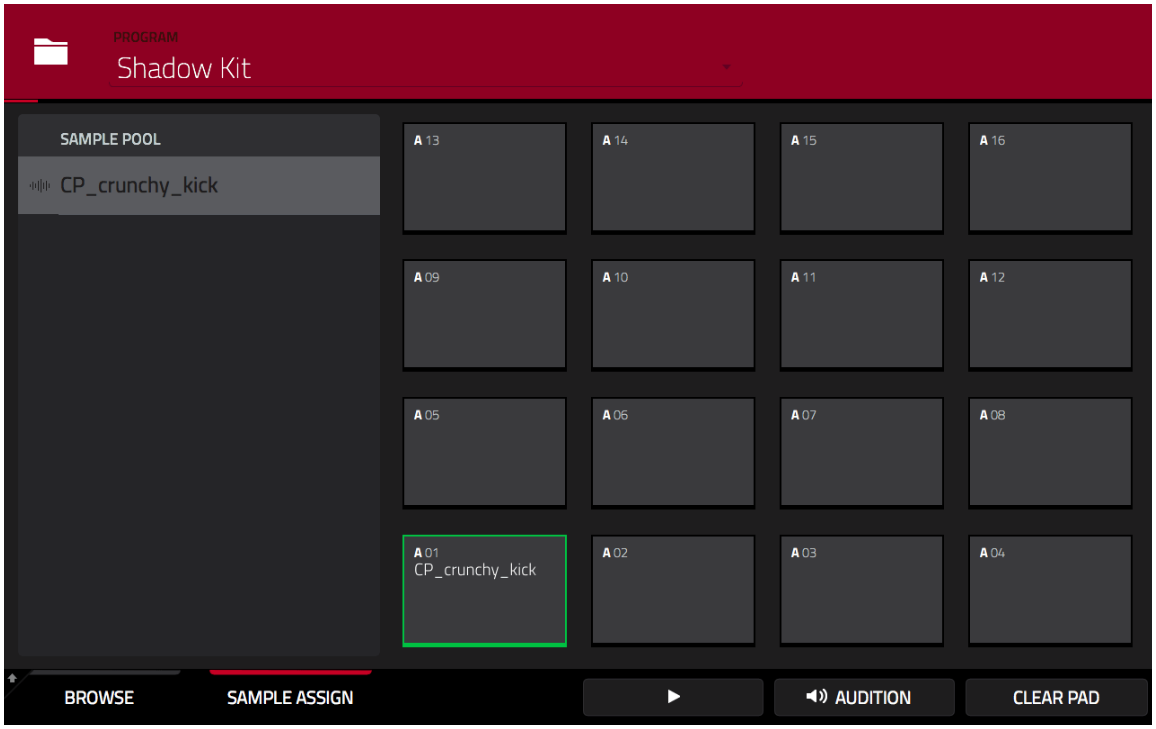
You can also route the audition audio through it's own dedicated outputs - configure this under the **AUDITION OUTPUTS** setting.

Locate the file '**CP_crunchy_kick.wav**' - let's assign this kick sample directly to **pad A01**. Do you remember we hit pad A01 previously to turn it green? This means that pad A01 is currently our target pad. Now when you load your kick it's going to get assigned to pad A01 automatically. *If A01 is not currently green, hit it again to select it.*

To load this kick you have a few options – choose any one of these:

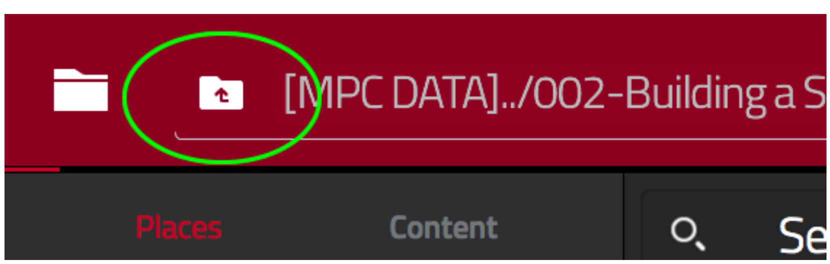
- Press the **LOAD** button (bottom right of touchscreen)
- Press the data wheel down so it 'clicks'
- Double tap the file name on screen

After loading, hit the **SAMPLE ASSIGN** button at the bottom of the screen:

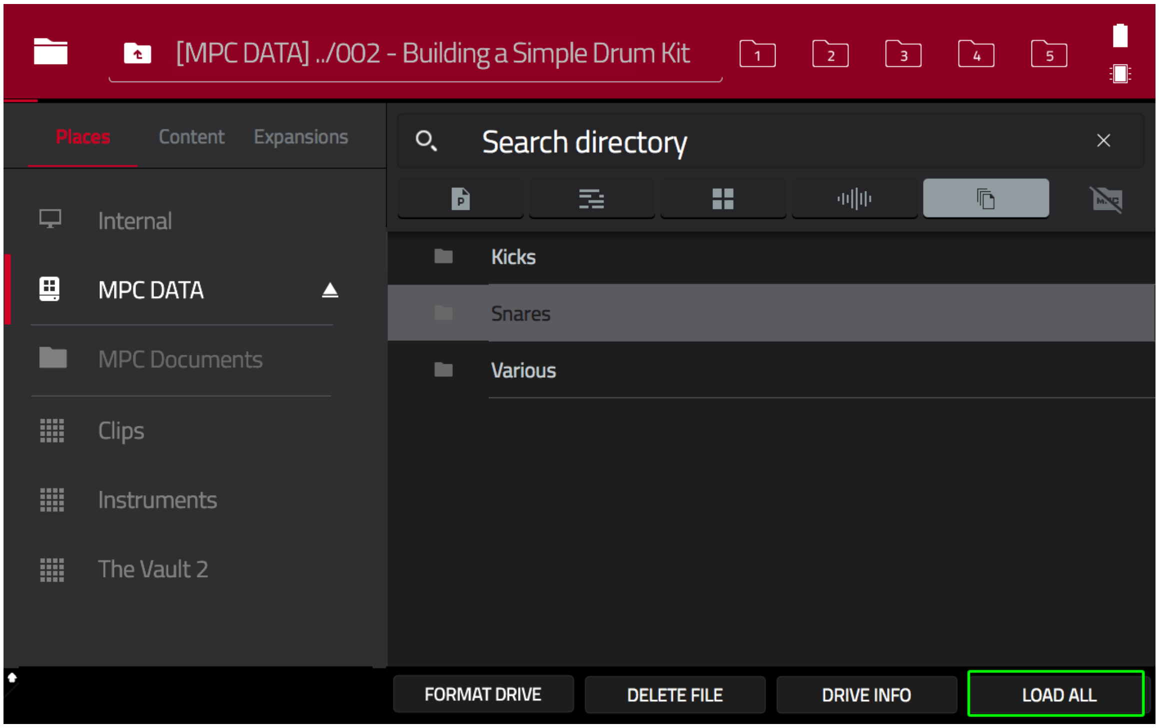


You can see that the kick sample is now assigned to pad **A01** in our DRUM program. Additionally you can see that the **CP_crunchy_kick** is now also added to our project's '**SAMPLE POOL**' on the left hand side of the screen. Hit pad **A01** to play your new kick drum pad!

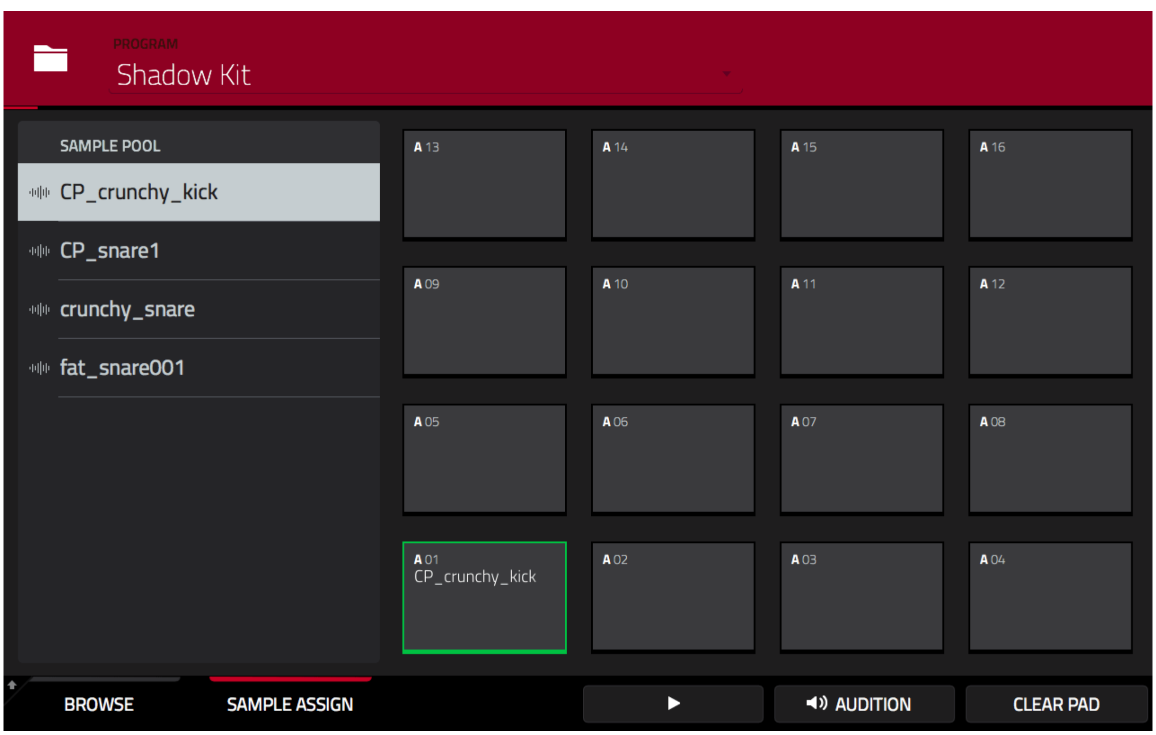
Tap the **BROWSE** button - you should still be inside the **Kicks** folder. Press the '**Up Directory'** icon to return to the **Chapter 2** folder:



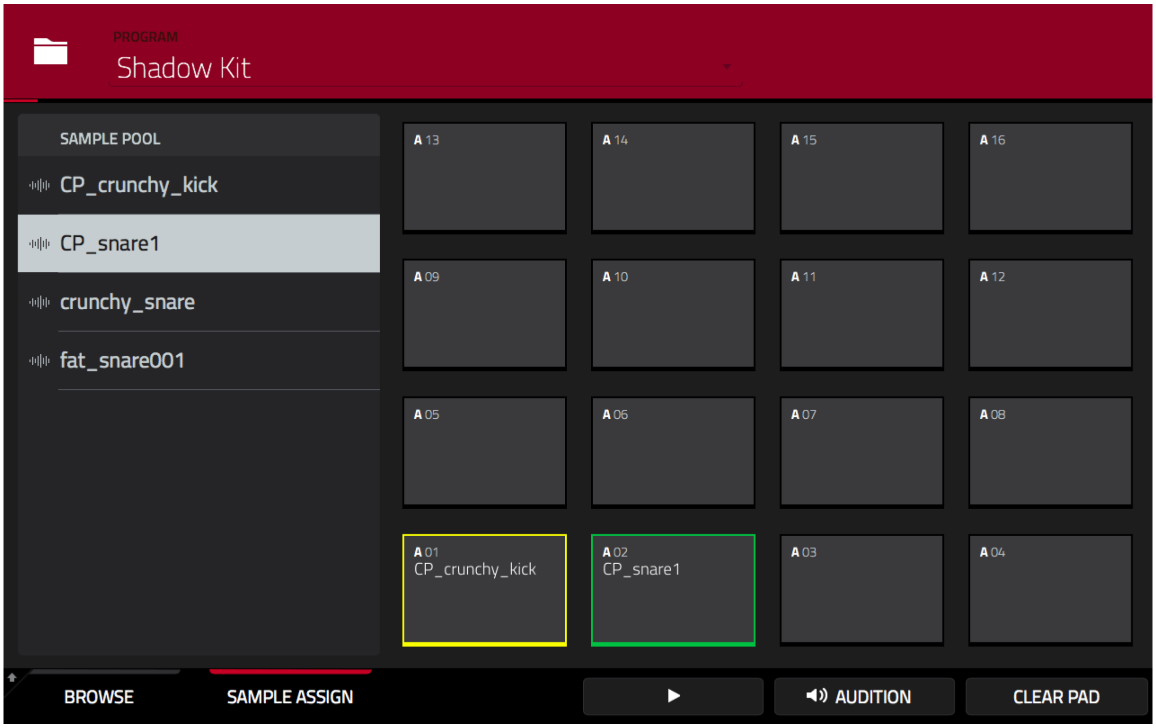
Now *single tap* the **Snares** folder to select it and hold down the **SHIFT** button. You'll see a new button appear at the bottom right of the screen; **LOAD ALL**.



With **SHIFT** still held down, press **LOAD ALL**. Release **SHIFT** and head back to the **SAMPLE ASSIGN** screen



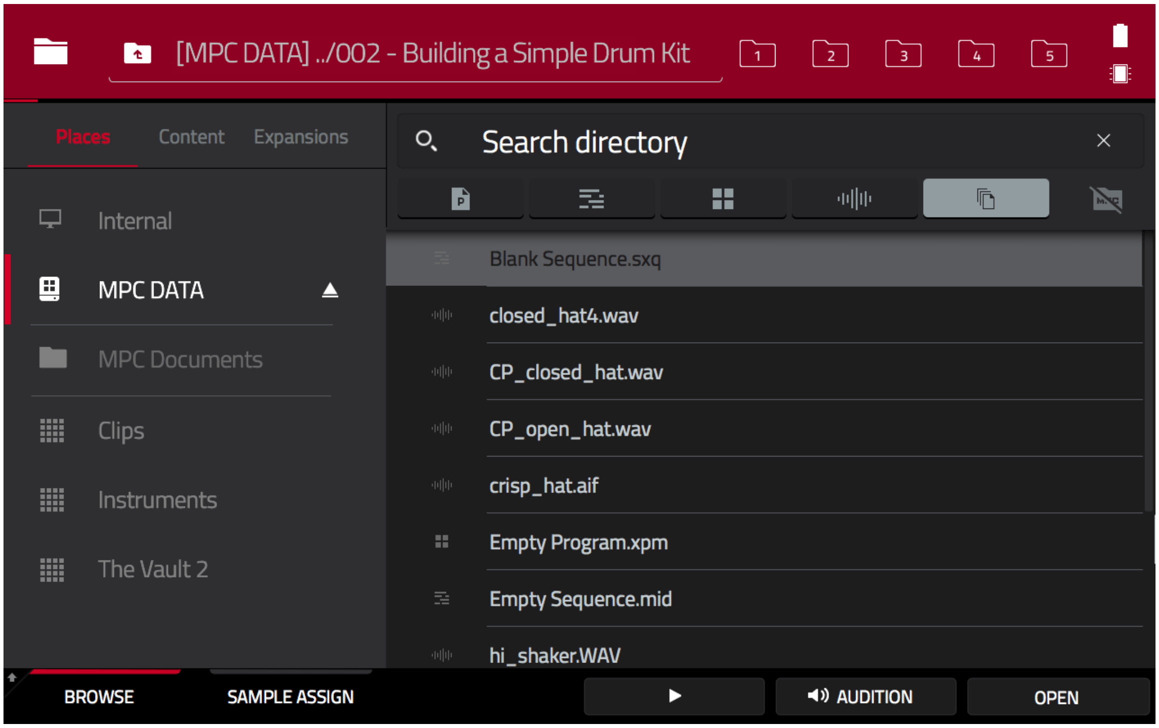
You'll now see three new snares have appeared in your **SAMPLE POOL**. Tap pad A02 to select it and use your data wheel to scroll and audition the new samples we've added (or just single tap each one). Assign **CP_snare1** to pad **A02**.



Play pads A01 and A02 to hear the foundations of your new drum kit forming.

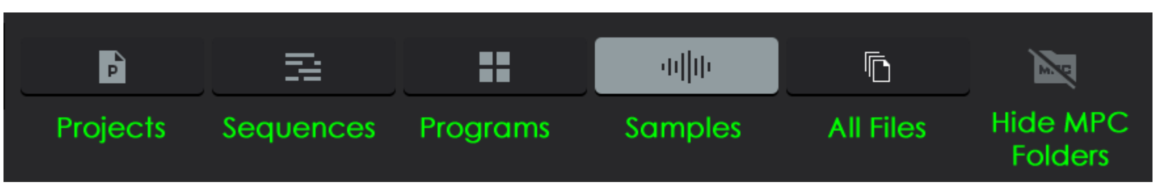
Browser Filters

Back in the **BROWSER**, leave the 'Snares' folder and go back up to the root of the **'002'** folder; enter the **Various** subfolder.

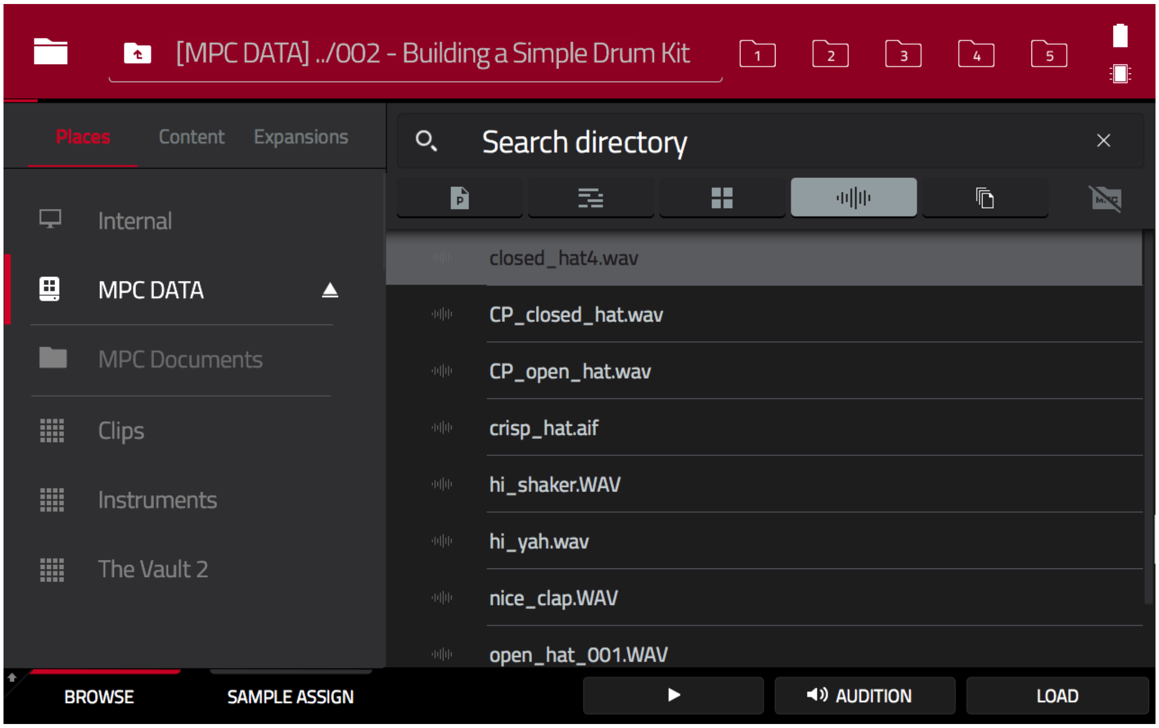


There's a whole bunch of different file types here, but we're only interested in loading **audio samples**, which are indicated by the following icon:

To only view audio files we can make use of the **File Type Filters** at the top of the file list.

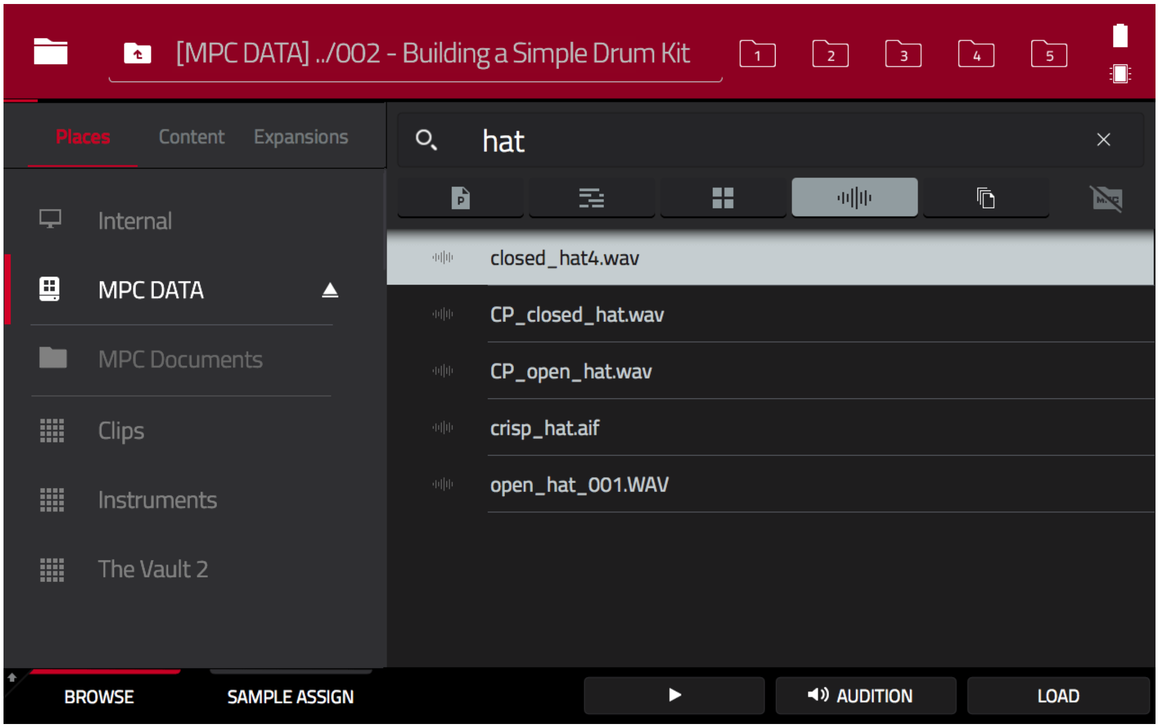


With a filter selected, only that particular type of file is displayed in the BROWSER. By default the filter is set to '**ALL**' (notice how it has a grey box around it), which means all files are shown. To view only samples, touch the '**Samples**' filter. You should now see the following:



This is a list of samples in the current folder; if there are more than eight samples in a folder you'll need to scroll downward to view them. This can be done with by turning the data wheel clockwise, using the '+' button, or dragging your finger upwards on the touch screen. You can increase the dragging speed by placing more than one finger on the screen as you drag. You can also skip quickly to the top or bottom of a list by performing a 2-finger long 'swipe' to the left or right.

You can also filter the contents of a directory using the '**Search directory'** box. This will only show files whose name contains the search string you enter. In the **Search Directory** box, enter '**hat'** so only samples with the term 'hat' in their file name appear:



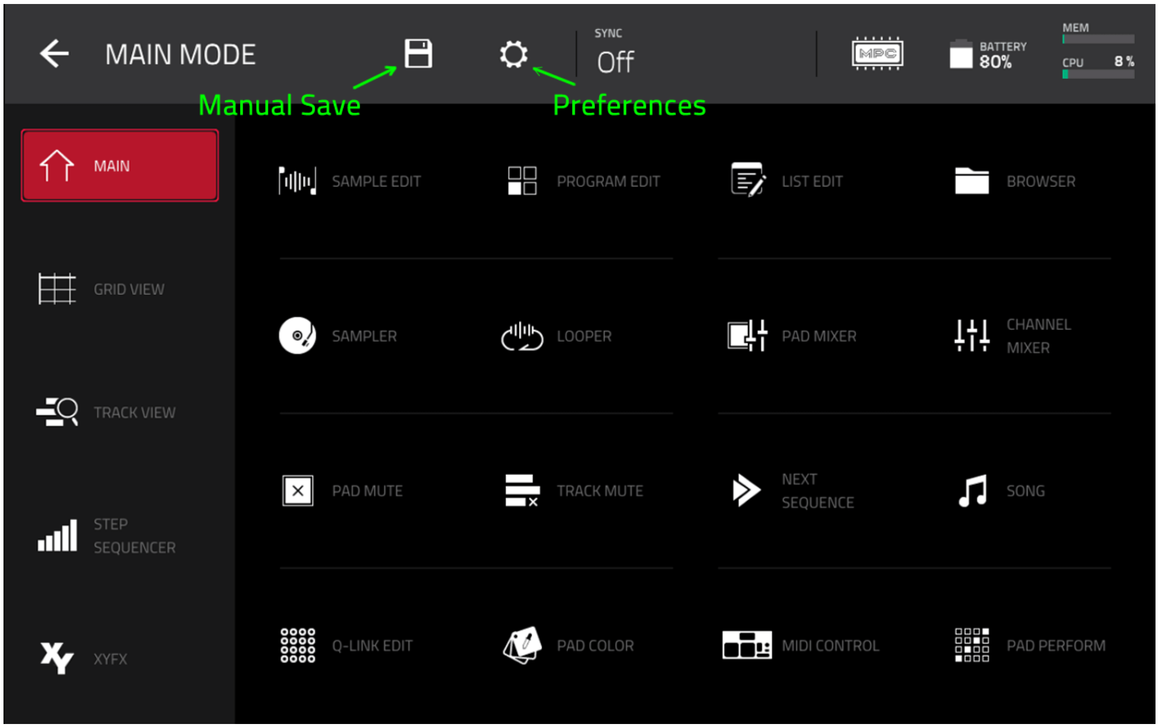
Hit pad **A03** so it turns green. Double tap the **CP_closed_hat** sample to simultaneously load and assign this closed hi hat to pad **A03**. Now hit pad **A03** to hear your hat within the kit.

Finally, assign the **CP_open_hat** sample to pad **A04**. You've now built a very basic drum kit – give it try and 'finger drum' a beat!

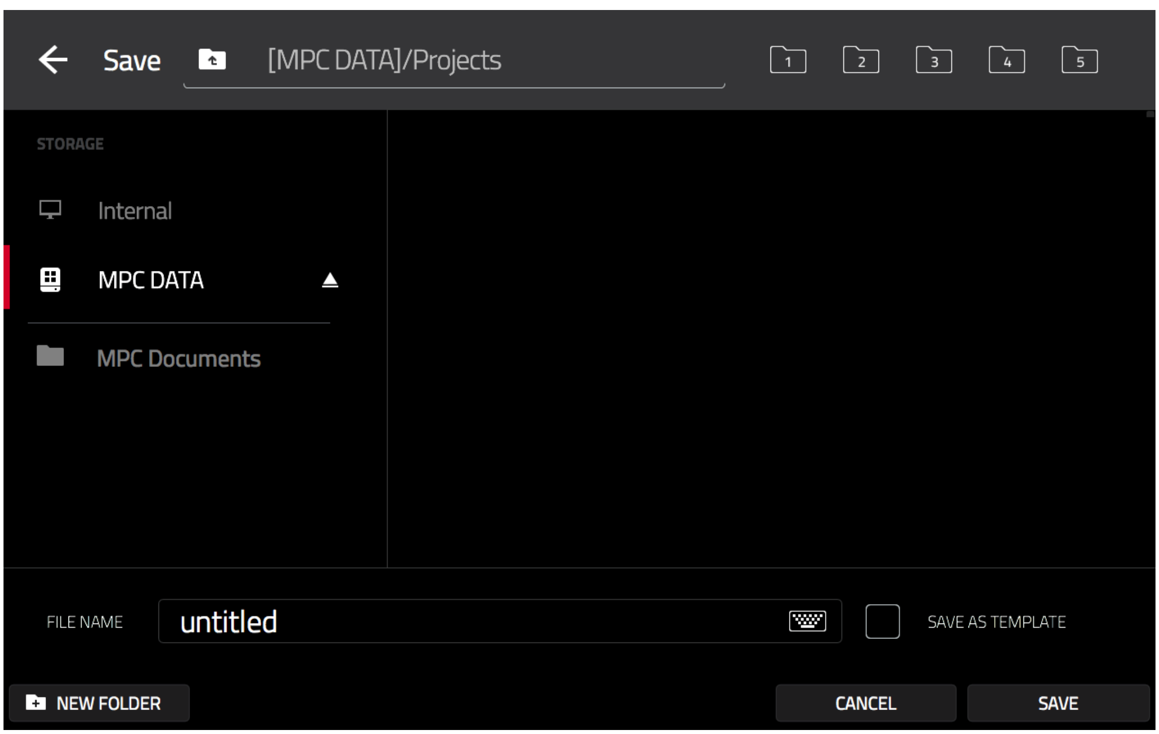
Saving Your Project

It's important that you save your work regularly. When you save your entire project you save *everything* currently associated with that project, such as programs, sequences, sounds, effect settings, mixer levels etc. A project file is a 'snap shot' of your entire project, so when you reload that project file, your entire project is recreated *exactly as it was when it was saved.*

One way to save your project is to head over to the **MENU** screen and press the **DISK** icon at the top of the screen:



In the MPC X, you can also hold down **SHIFT** and hit the **BROWSE** button. The first time you save a project you'll see the following screen:



Here you can choose a 'save' location and a project name. Now, while it is possible to save to the internal 16GB drive (Akai would prefer you to save to the

'MPC Documents' folder), it is very important to note that currently, Akai have provided no means of *externally* accessing the 16GB internal drive; you cannot see it in 'controller mode' nor can you move files to it (or from it) when in standalone mode.

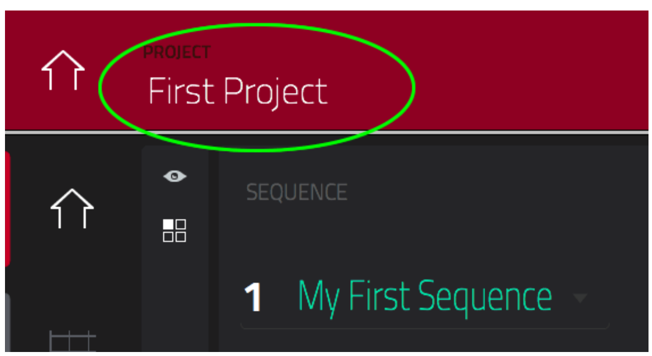
This means you cannot backup or externally access any projects you save to this drive!

Hence I recommend you save all your projects to an 'attached' drive. This could be an SD card, a USB drive or an internal SATA drive (if you've installed one). Please see **Appendix B** for more information about these issues.

Select an attached drive and choose a save location – in the above screenshot I have chosen the '**Projects**' folder on my **MPC DATA** disk. Tap on the current default project name '**untitled**' and use the on screen keyboard to enter the name **'First Project'.**

To simultaneously rename and save your entire project, tap the **SAVE** button at the bottom of the screen.

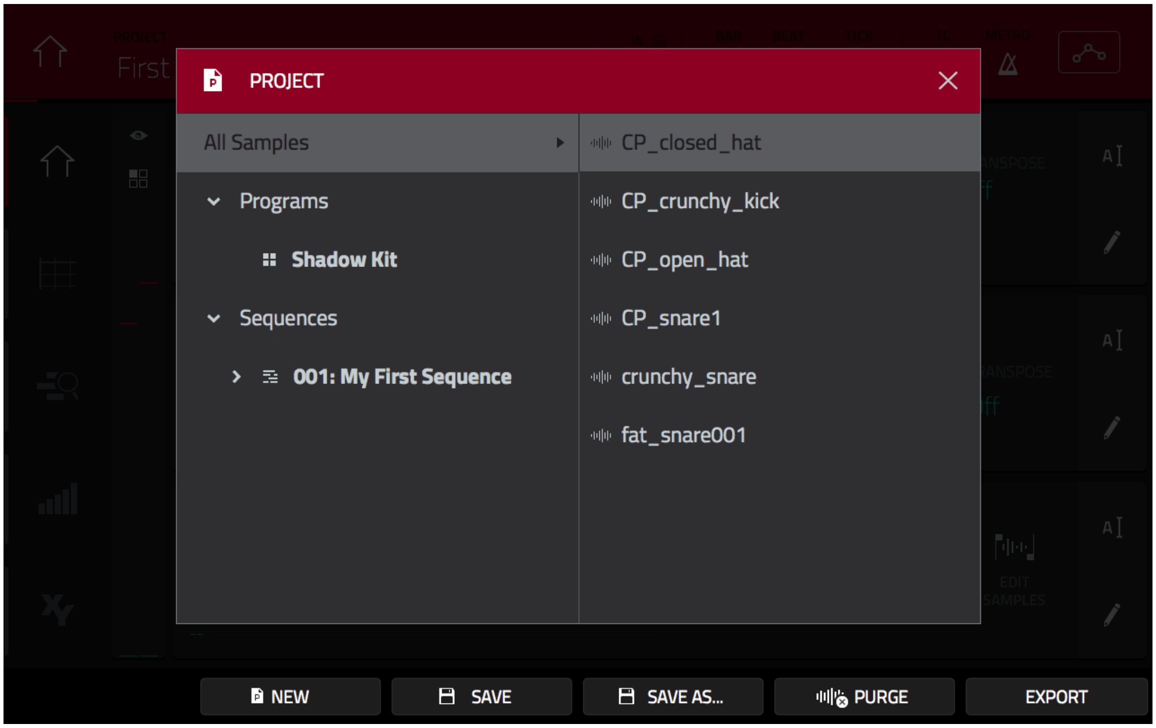
Go to **MAIN** and you should now see the name displayed at the top of the screen:



Now that your project has been initially saved you can quickly re-save it by either returning to the **MENU** page and hitting the disk icon, holding down **SHIFT** and **BROWSE** (MPC X only) or another option is to save via the **MAIN** screen.

In **MAIN**, hit the '**folder**' icon in the top centre of the screen:

This brings up the **PROJECT** panel.



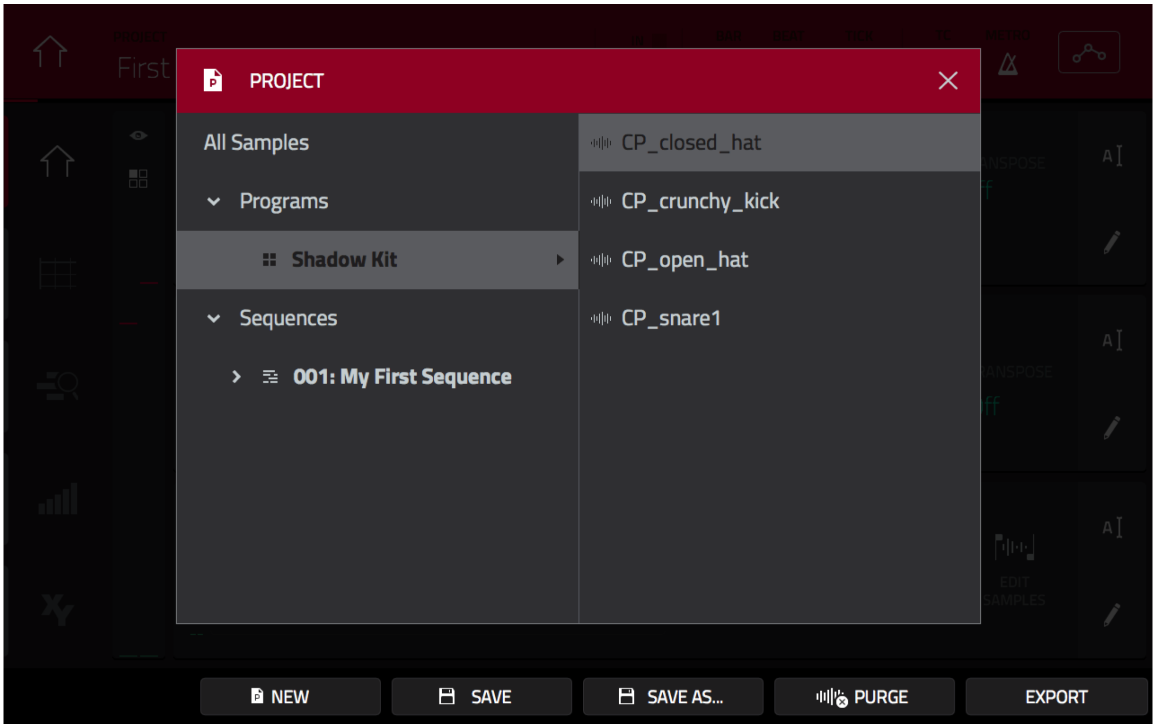
The PROJECT panel shows a snap shot of all the resources in your project. As you can see it also contains a SAVE option at the bottom of the screen; there is also a 'SAVE AS' option which gives you the opportunity to save your project under a new name or to a different location.

Moving Internal Drive Projects To External Disks

The **SAVE AS** option is currently the only way you can 'backup' any projects previously saved to the internal 16GB drive. To do this, load up the project into active memory, select SAVE AS and choose an attached disk as the 'save' destination. This will save a copy of your project to this other disk which can then be accessed outside of your MPC (see Appendix B for more information on how to do this).

Project Filters

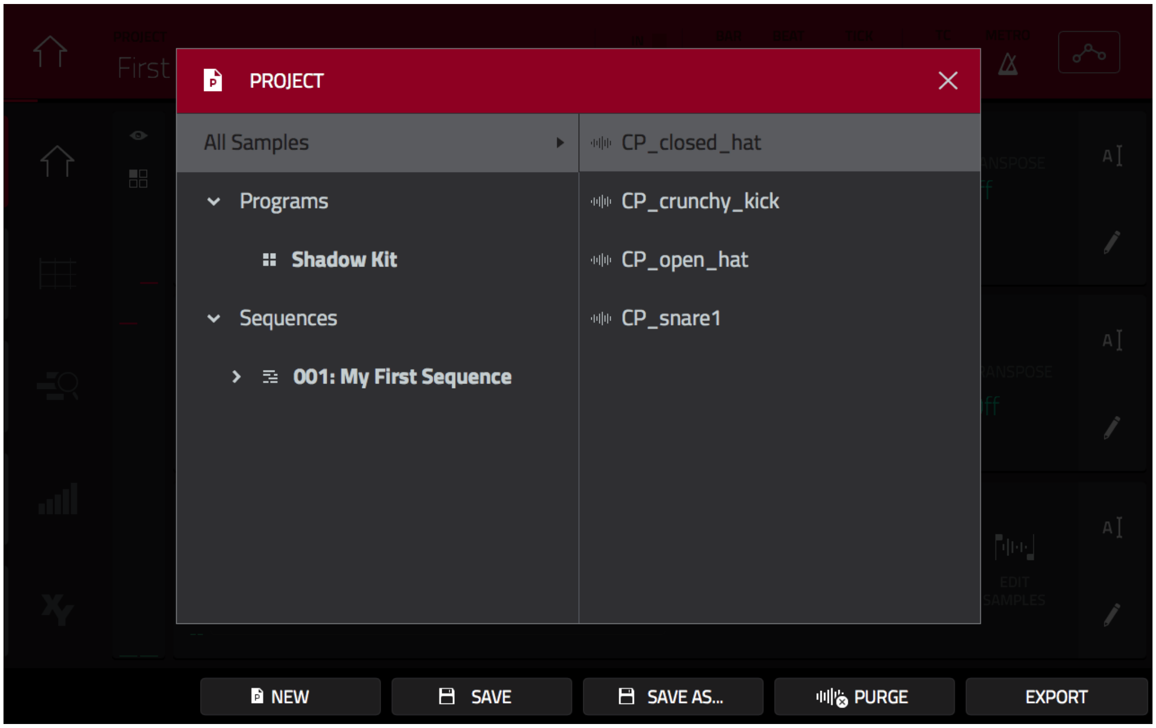
With '**All Samples**' selected on the left side of the panel, the right hand side shows all six samples that have been loaded into your project. Now tap the **Shadow Kit** program entry:



The **PROJECT** panel now shows us the four samples currently assigned to this program only.

Freeing Up System Memory (Purging)

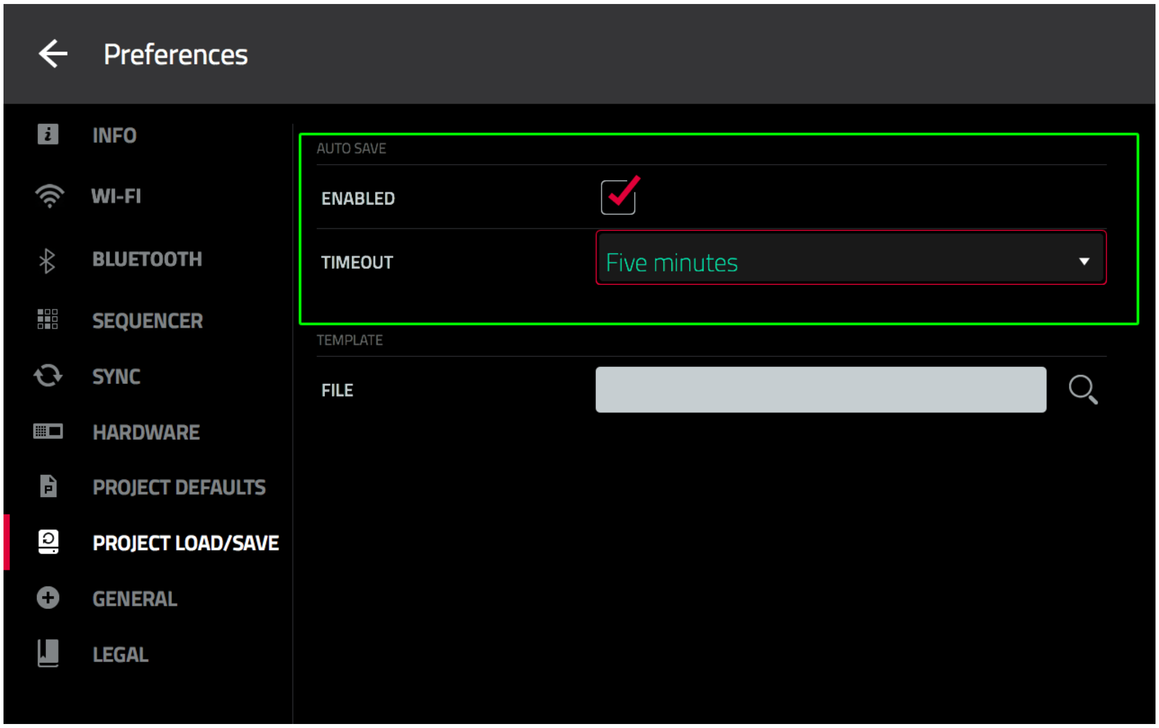
Select **ALL SAMPLES** again. Now hit the **PURGE** button and select **UNUSED SAMPLES** - this completely removes all unassigned (i.e. unused) samples from our project and subsequently frees up that system valuable memory.



Now those unused snares you loaded have disappeared from your project. Purging is the fastest way to rid your project of unnecessary clutter; just remember that purged samples cannot be retrieved, so only purge sounds that you know you don't need anymore.

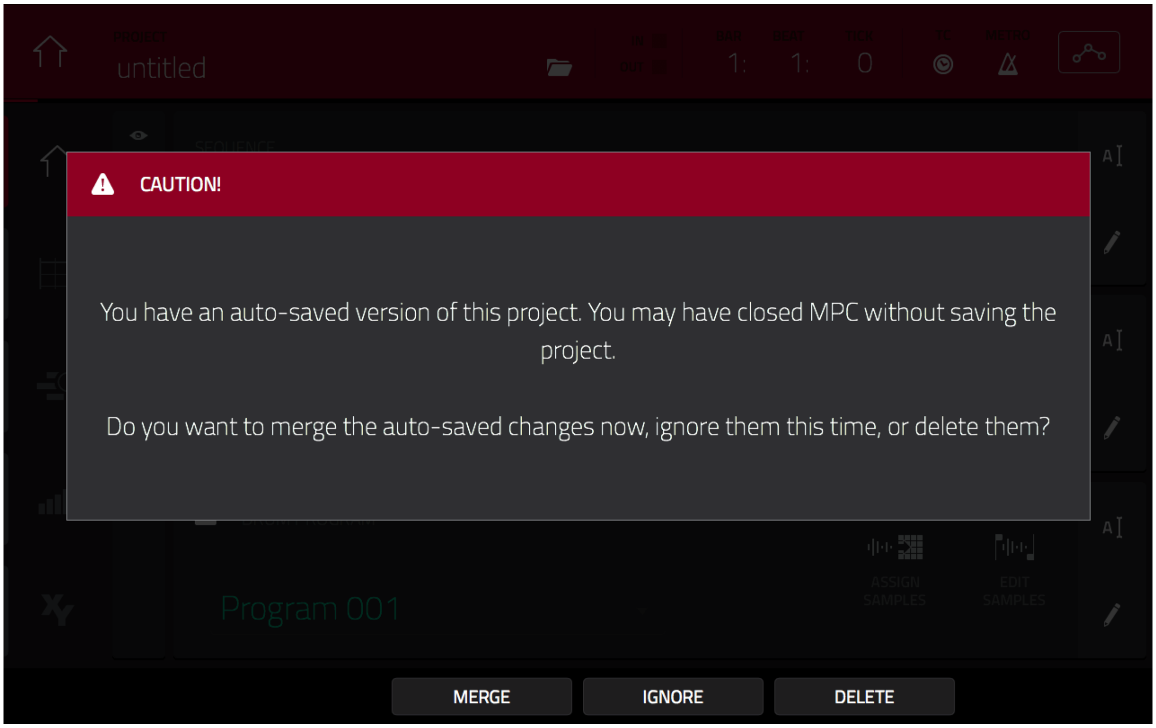
Auto Save

Head back to **MENU** and go to **PREFERENCES** > **AUTO SAVE** from the left hand options:



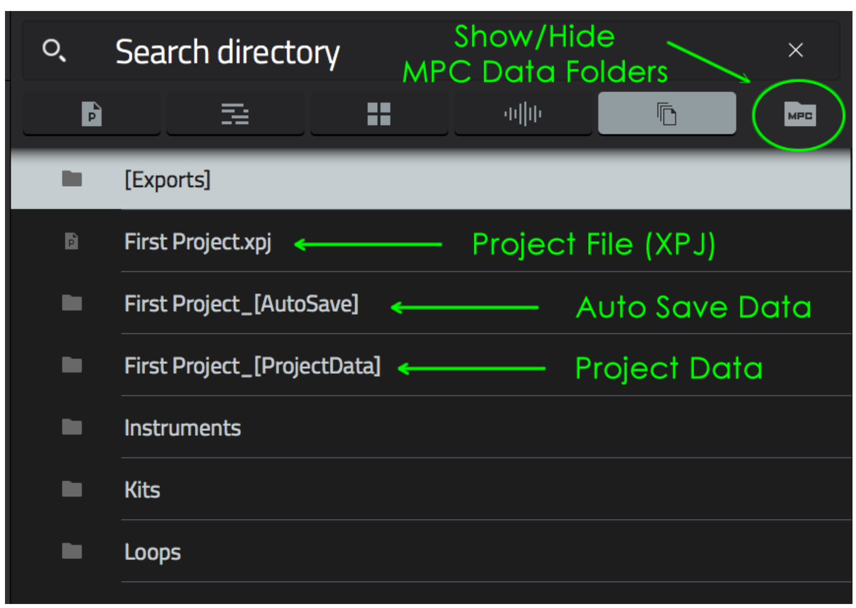
AUTOSAVE will automatically save your entire project at regular intervals. Check the '**ENABLED**' box and set a **TIMEOUT** – this is the interval the MPC waits between autosaves. If there's been no changes, it will not bother saving.

If you have a system crash, the next time you try to load the project that was open at the time of the crash, you should see the following:



Here you can merge the autosave and standard versions of the project, ignore the auto saved version or just delete it entirely.

You can also access autosave versions directly from the BROWSER. 'Auto saves' are made to a separate folder in the same directory as the project you are backing up:



Inside the [**AutoSave]** folder you'll find the latest autosaved copy of your project (i.e. the project's XPJ file and the project's 'data' folder which contains all the sounds, programs and sequences used in the project).

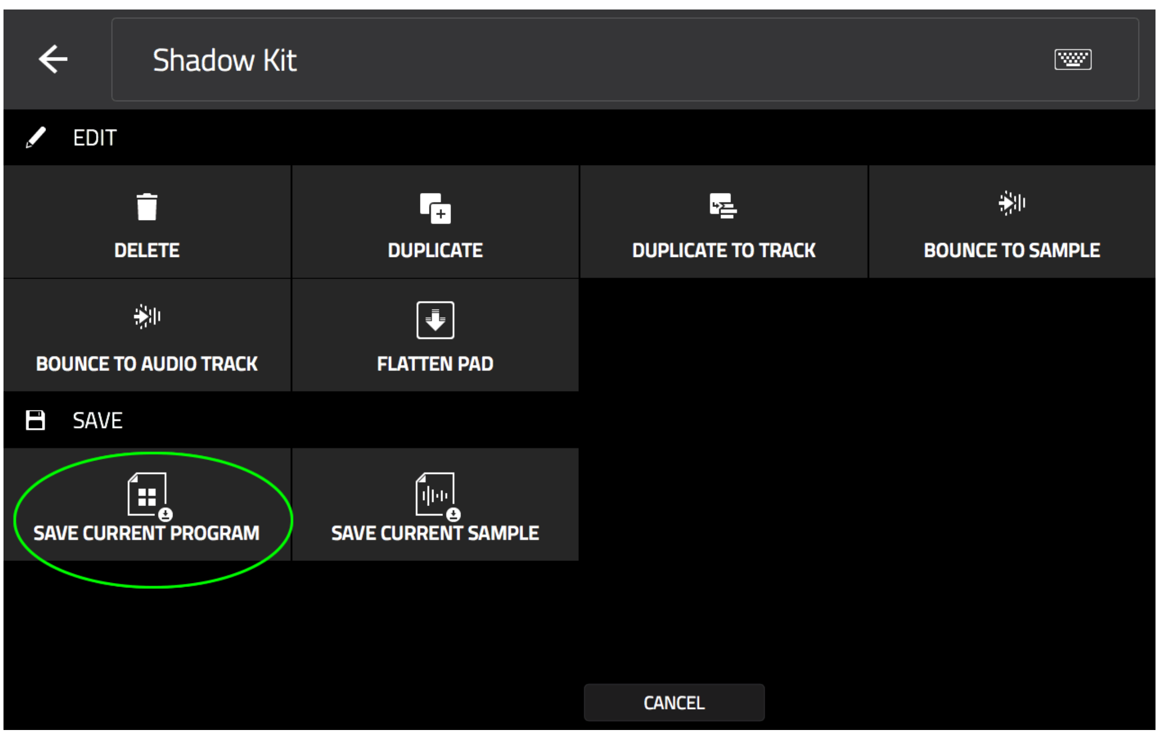
So with autosave you always have two concurrent copies of your current project. If you somehow manage to mess up your main project you can go inside the [AutoSave] folder, load up that one and hopefully it will contain a version of your project before you messed it up.

*Can't see your autosave or project data folders? Make sure the '**MPC**' icon at the far right of the filter icon list is active as this toggles their visibility.*

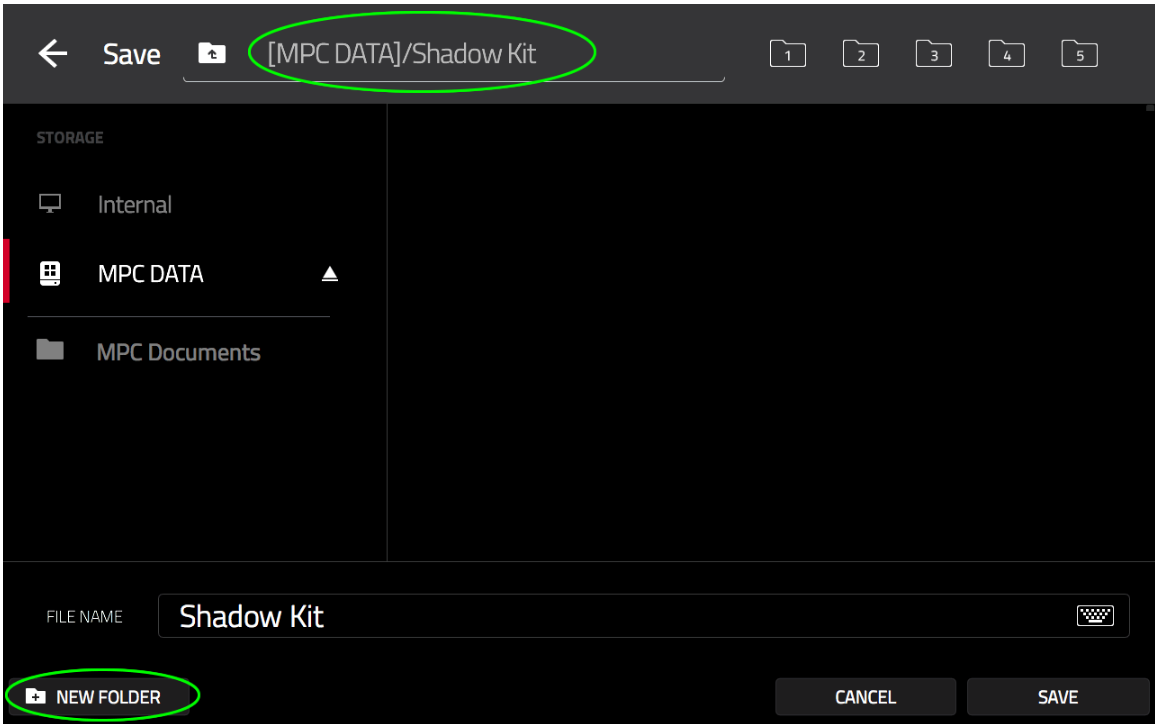
Saving Individual Programs

Rather than save your entire project, you can also separately save an individual program. This might be handy if you wished to distribute one of your kits online or with friends.

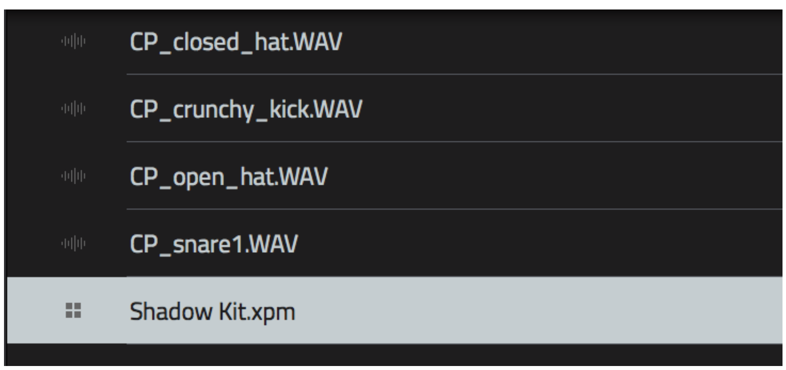
Go to **MAIN** and on the **DRUM PROGRAM row**, hit the **pencil** icon:



Simply select '**SAVE CURRENT PROGRAM**' and choose a location to save the program to. Ideally this should be within a subfolder on one of your removable disks (e.g. SD card or USB disk). Use the '**NEW FOLDER**' option on the 'Save' screen to create the subfolder and give it a name to match your program name (e.g. '**Shadow Kit**').



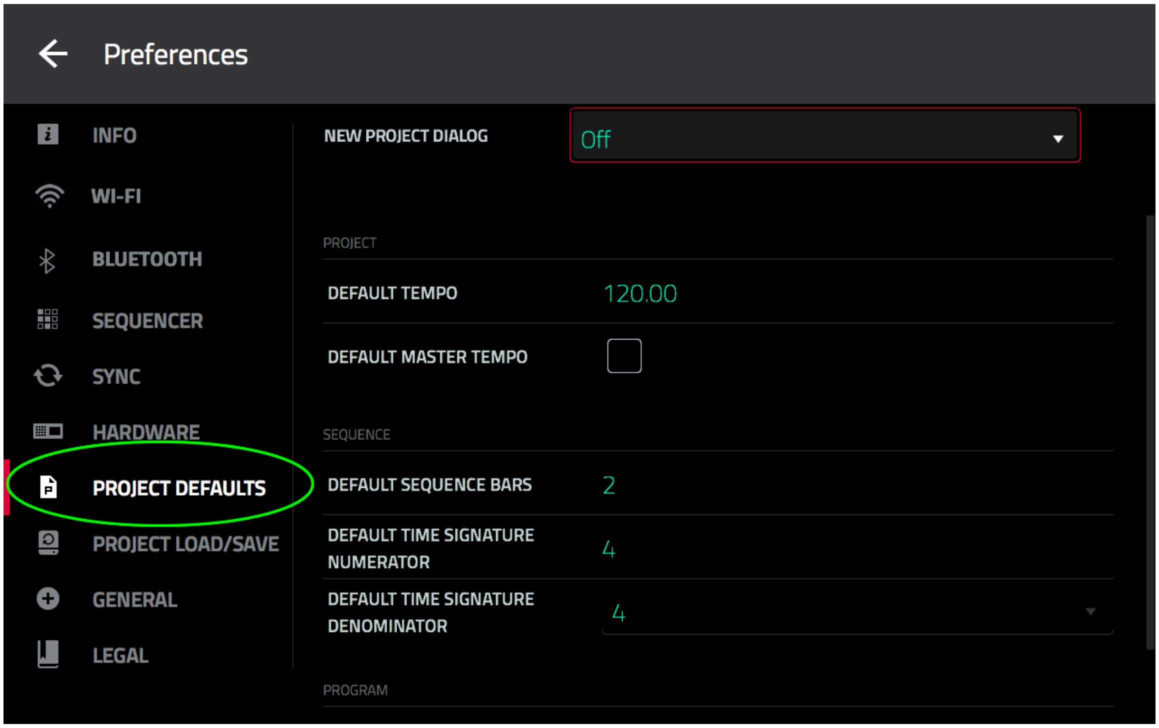
After 'saving' a program the MPC will have copied everything needed to recreate that program inside the target folder - i.e. all the samples used in the program and the actual '**XPM**' program file itself.



It's vital that you don't move or rename any of the samples in that folder - everything needs to remain where it is, otherwise the kit will no longer load correctly. When you distribute your kit, simply distribute the entire folder.

Default Project Settings & Project Templates

As you work through this book and begin developing your now unique workflow it is likely that you'll begin to prefer certain project specifications available in every new blank project. Some default project variables are set in the **PREFERENCES** screen under **PROJECT DEFAULTS**:



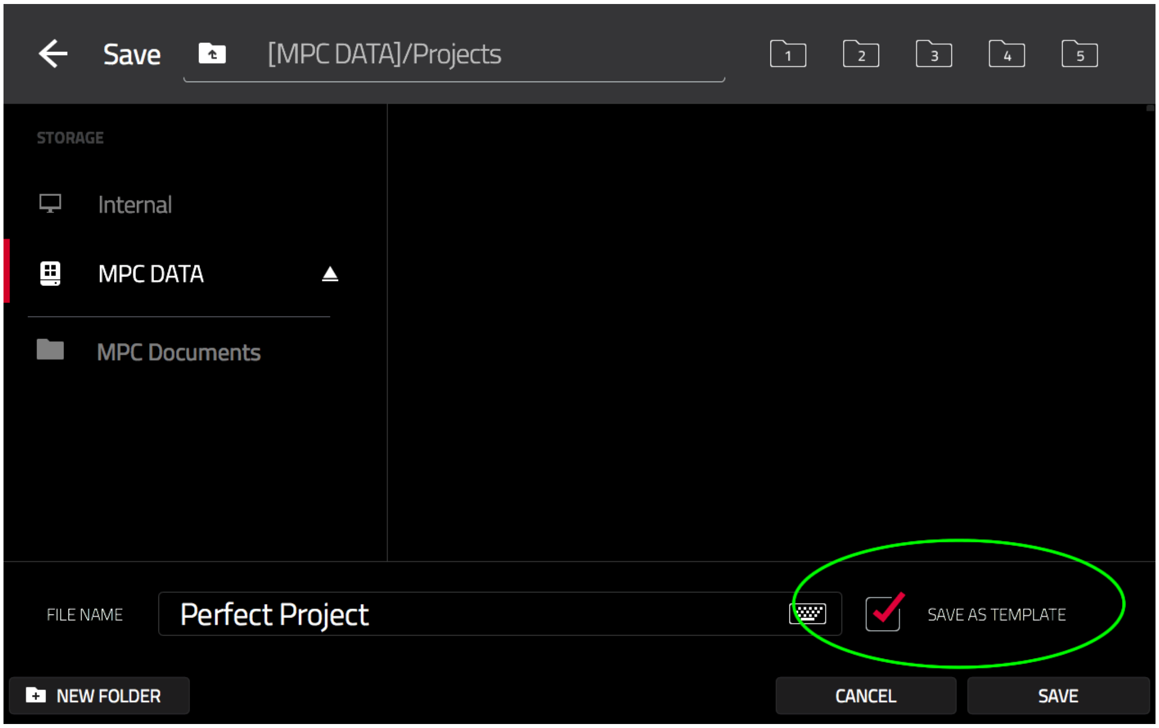
This screen allows you to configure basic things like a default sequence tempo and bar length. However if you wish all your default projects to have a more specific and complex set up, then you'll need to create a **Project Template.**

A Project Template is simply a saved project structure that has all your favourite settings pre-configured. For example, you might like all your projects to have a baritone sequence structure, say 4 basic sequences (e.g. intro, chorus, verse, ending), so your template will have these four sequences already set up.

If you always use the same track structure then each sequence can set up with those tracks pre-configured, for example track 1 (kick), track 2 (snare), track 3 (hat), track 4 (bass) and so on. You can even have your template pre-load your favourite kits and instruments and have them already assigned to your tracks. If it's configurable, then it can be saved in your template - Q-LINK assignments, audio output assignments, MIDI channels, effects, etc.

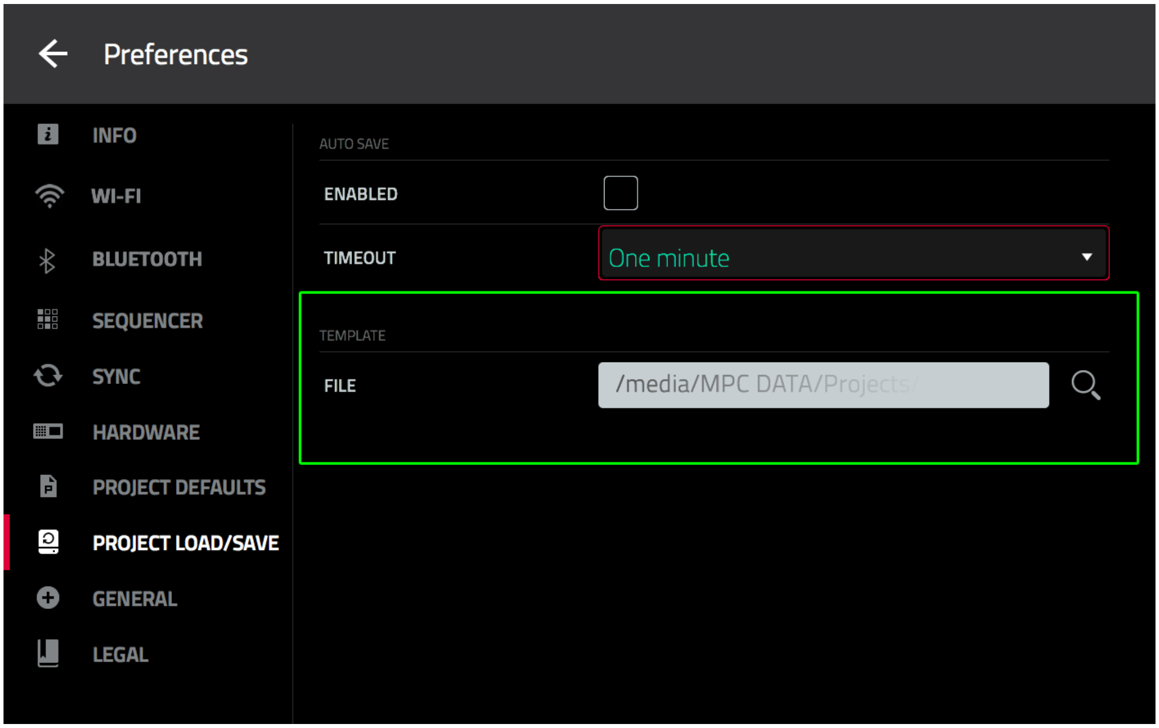
It's early days in our MPC journey so it's unlikely you'll be ready to create a template just yet, but when you are ready to do so the process is very easy.

Simply set up the project you wish to save as a template, go to **MENU > SAVE** and choose a name and location (as described in the previous section), but before confirming the save, make sure you check the **'SAVE AS TEMPLATE'** box.



Hit **SAVE** to save this project as a template. You can save a template to any location including external disks, although if the disk is not present that template will not be available for selection.

You can only have one project template configured at any one time, but you can easily change it by either re-saving a new one or going to **PREFERENCES > PROJECT LOAD/SAVE**:

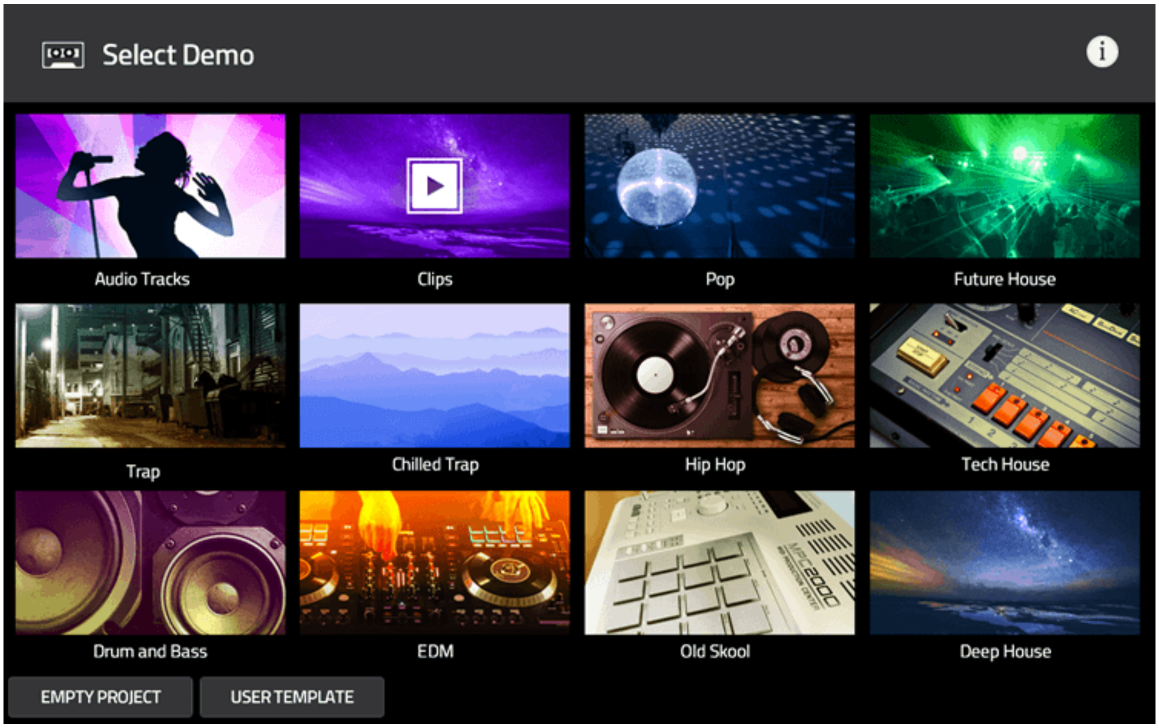


Simply browse to a new project via the '**FILE**' option under **TEMPLATE**.

Creating a New Project

In chapter 1 we configured the MPC to boot up directly into a blank new project by setting **PREFERENCES > PROJECT DEFAULTS > NEW PROJECT DIALOG** to '**OFF**'. However there are two other options available here; '**Demo**' and '**Demo/Template/Recent**'.

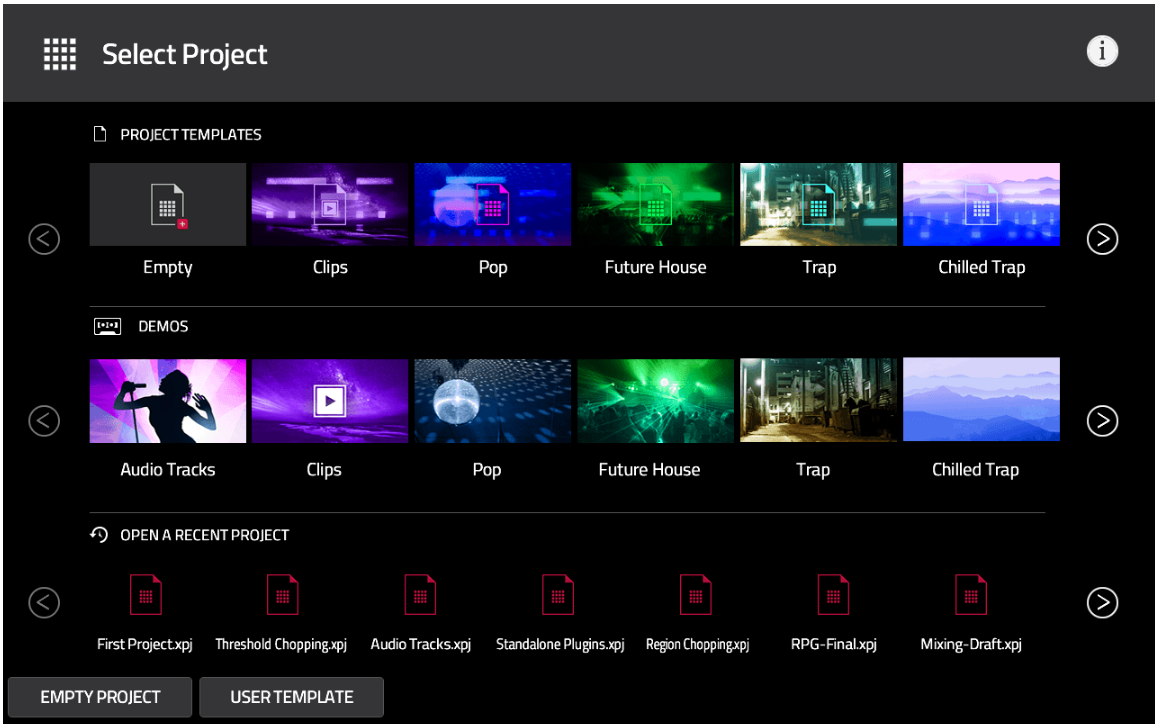
With '**Demo**' set, upon boot up you are presented with an initial '**Demo**' landing page:



Here you can select a demo project that loads a small demo song in a particular style - simply tap on the demo and it immediately loads up. This is a great way to analyse how other people set up and build their songs. You also have an option at the bottom left of the screen to select an '**EMPTY PROJECT**'. If you tap this your MPC will open a a blank project and take you directly to MAIN.

In my opinion the '**Demo**' option is fairly pointless as you can load the demos at any time directly from the **BROWSER** - simply select the '**Content**' tab on the left of the screen and tap the '**Demos**' icon.

A more useful option is '**Demo/Template/Recent**'. With this set you are presented on a slightly different screen upon boot up:

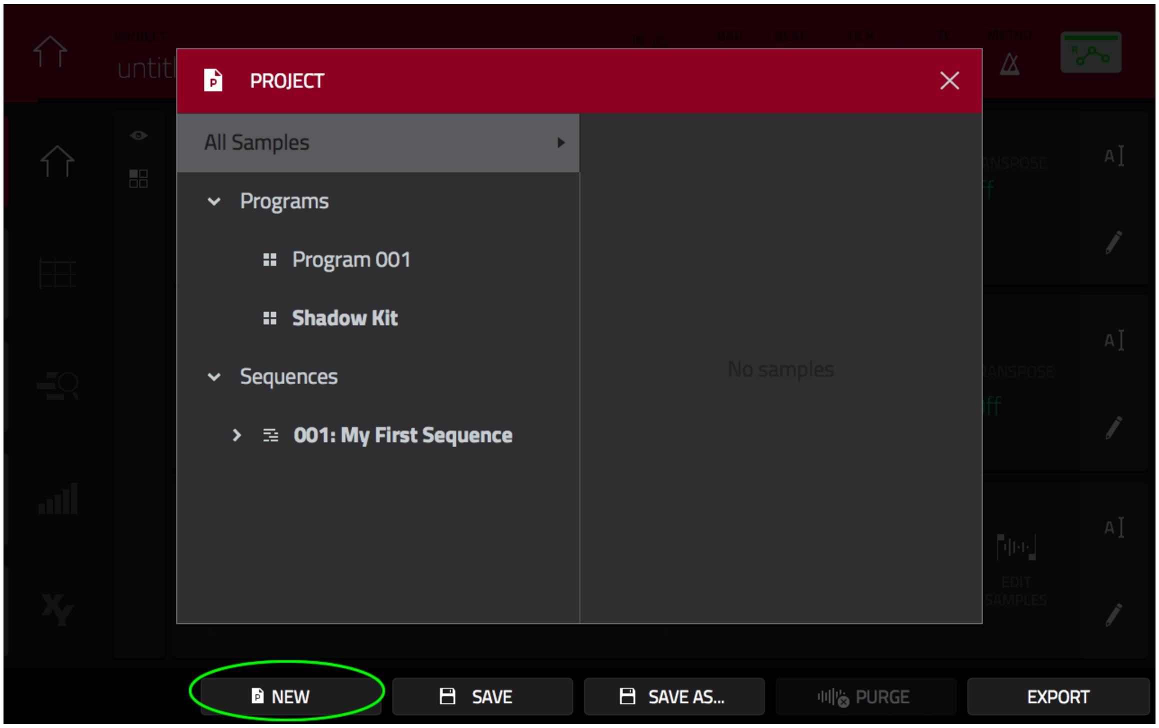


This is the '**Select Project**' screen. Here you can choose to load the standard demos (as before) but you can also pre-configured '**project templates**' that are basically the project demos without any of the sequence data pre-loaded, giving you a starting block to build your own beat in a particular style.

A third option is to load a '**recent project**', which shows a list of your own user projects that you've worked on most recently, which is handy for jumping directly back to the last project you were working on in your previous session. This boot up screen also has the option of loading a standard '**EMPTY PROJECT**'.

Additionally on both of these screens if you previously created a 'Project Template', then this will be available to load at the bottom of the screen by tapping the '**USER TEMPLATE**' button (otherwise this option is greyed out).

Other than a fresh boot up of your MPC, the other way to create a new blank project is via the **PROJECT** panel.



Simply select **NEW** at the bottom left of the screen. This option also uses your '**NEW PROJECT DIALOG**' setting, so for example, if you have set this to '**Demo/Template/Recent**' then hitting **NEW** from the **PROJECT** panel will display the 'Demo/Template/Recent' landing page screen. If you set 'NEW PROJECT DIALOG' to 'Off', hitting new will just immediately create a new blank project.

When creating a new project from the PROJECT panel, your existing project is completely wiped from active memory, but your MPC will prompt you to save the existing project before doing so.

003 Recording Your First Beat

In this tutorial we're going to use our basic kit and record a drum beat to our blank sequencer track.

There are many ways in which we can create a sequenced drum beat in the MPC MIDI sequencer and in this section we'll take a look at standard '**real time**' recording where we simply record our events 'live' into the sequencer using the pads.

Real Time Recording Configuration

In this example I want to capture every nuance of the performance exactly, so let's turn **FULL LEVEL** 'off' by pressing the **FULL LEVEL** button on your controller so it is no longer illuminated. With FULL LEVEL 'off', the MPC will record the velocity (hardness) of each pad hit just as you originally played it (with FULL LEVEL: ON, the sequence records every single pad hit at full velocity, even the lightest touches).

Go to **MAIN** and make sure **sequence 1** (**My First Sequence**) is selected. This is the sequence we set up in chapter 1. To recap on how this is configured:

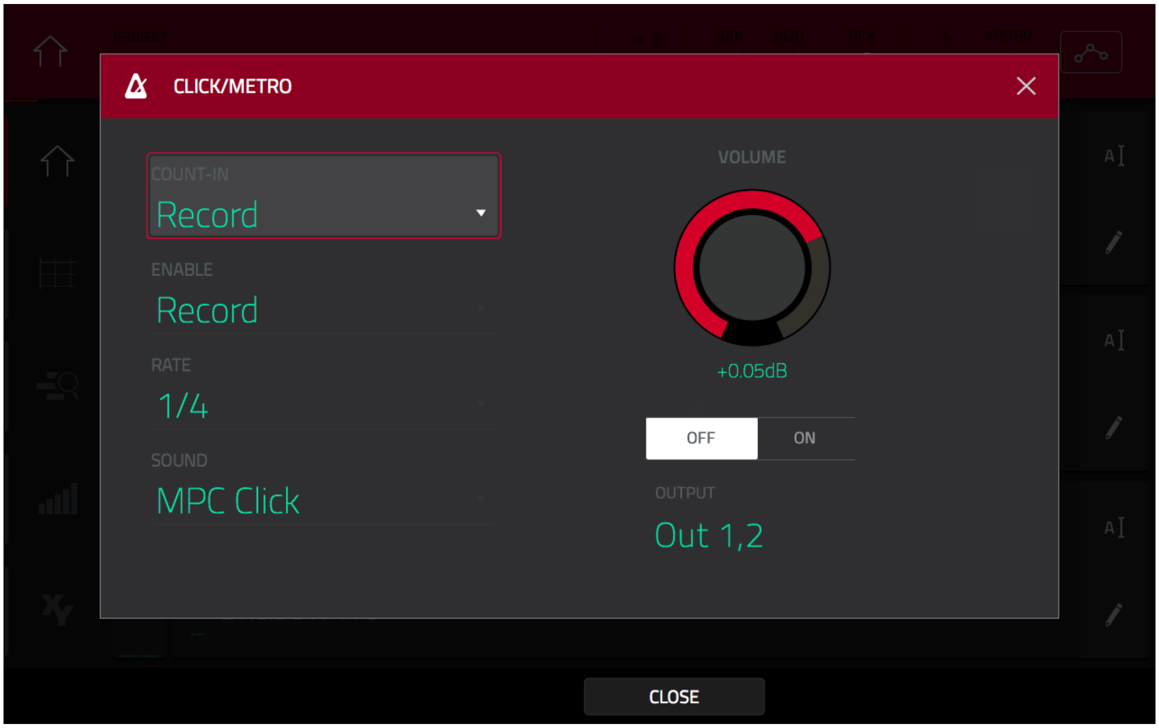
- The sequence has a **tempo** of **90BPM**
- We have set the sequence **LOOP** '**off'** so we can record our sequence for as long as we wish
- We have turned both **Timing Correct** (**T.C**.) and **FULL LEVEL** '**off'** to capture our performance exactly as it is played
- We're going to record our drums to **track 1** (**Drums**).
- Our basic drum kit program '**Shadow Kit'** is assigned to track 1.

In addition to the sequence-specific settings, you'll also want to set up your MPC's *metronome.*

Metronome Settings

The MPC has a built in **metronome** which outputs a click throughout your sequence to ensure you keep the correct tempo when recording a live performance.

Press the **METRO** icon at the top right of the touchscreen:



The ‘**ENABLE**’ option lets you choose when or if the metronome plays. For most situations I feel it only makes sense to have it playing when recording, so I set it to ‘**Record**’.

The ‘**COUNT-IN**’ option controls whether or not you’ll hear a 4 click count in before recording or playback starts. This can be vital if you intend playing events at the very start of a sequence. I personally prefer to have this only when recording, so I set mine to ‘**Record**’.

The ‘**RATE**’ option controls how often you’ll hear the click. For most uses, the default **1/4** (i.e. a click every 1/4 of a bar) is typically perfect, but you can increase the rate whenever you feel it’s required (we’ll see what all these different rates mean later when we look at timing correct options).

The ‘**SOUND**’ option controls the actual sound used for the metronome. As a long time hardware MPC user I love to hear the familiar classic MPC metronome sound, so I leave this set to ‘**MPC Click**’.

The on-screen dial is simply a volume control for the metronome – set this to any value you prefer.

OUTPUT lets you route the metronome through a specific set of outputs. This is very handy during live performances and recordings should you need to play to a click track as it allows you to route the metronome to a separate monitoring system which for example, can only be heard via headphones. You can choose a stereo pair (e.g. ‘3,4’) or just a single mono output (e.g. ‘3’).

Recording Your First Take

So with everything configured, let’s just get something recorded! If you can finger drum, that’s great, but if you can’t, who cares? Just play pads **A01**, **A02**, **A03** and **A04** in any way you wish, just so we can get some events recorded to this track.

To record events to the currently selected track in real time, first press the red **REC** transport button on your MPC.

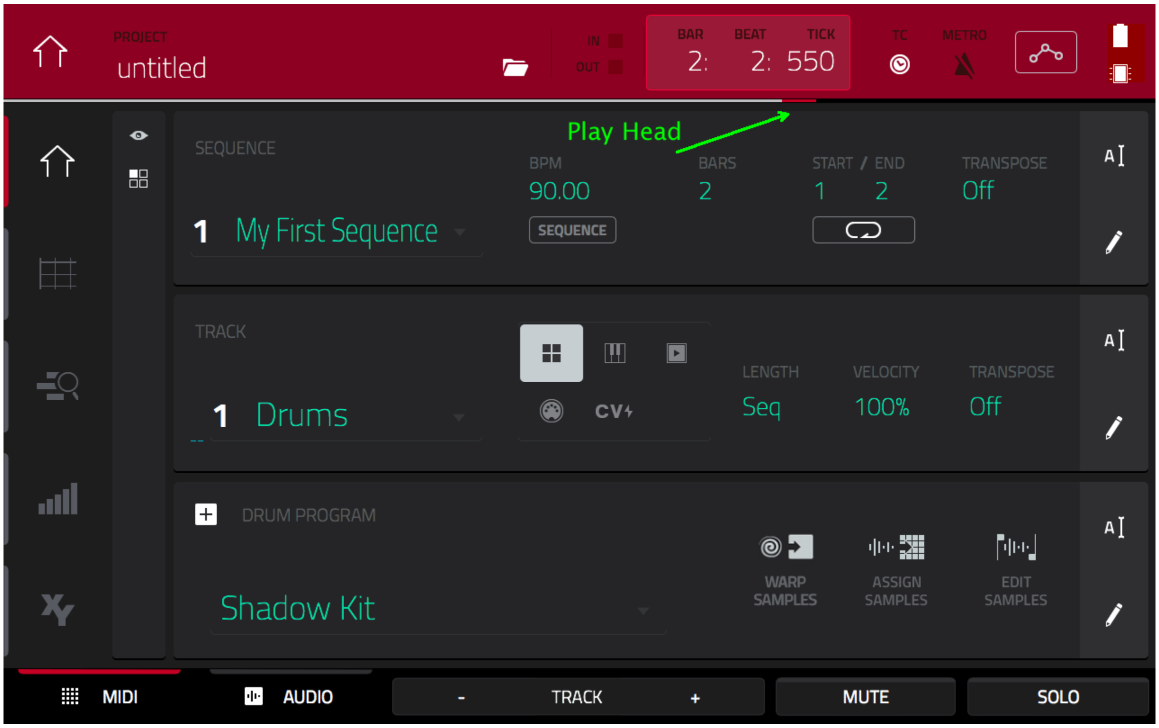
Upon doing this, the PLAY button will begin to blink, indicating that the MPC is now ‘armed’ and ready for recording. To begin the recording process, simply hit **PLAY START**.

PLAY START or PLAY?

The **PLAY START** button will always play your sequence from the very beginning of that sequence, while the **PLAY** button begins playback from where the current time marker is set (typically from the place you last pressed STOP).

After your count-in, you can start to play your pads and the events will be recorded. As you set **LOOP** to **OFF**, the sequence simply carries on recording past 2 bars and just continues to record the events you play via the pads.

During recording and playback of a sequence you’ll notice a white and red bar run across the screen:



This is the sequence 'play head', or 'time location indicator' and simply shows you whereabouts you are currently located in your sequence (more on sequencer time later).

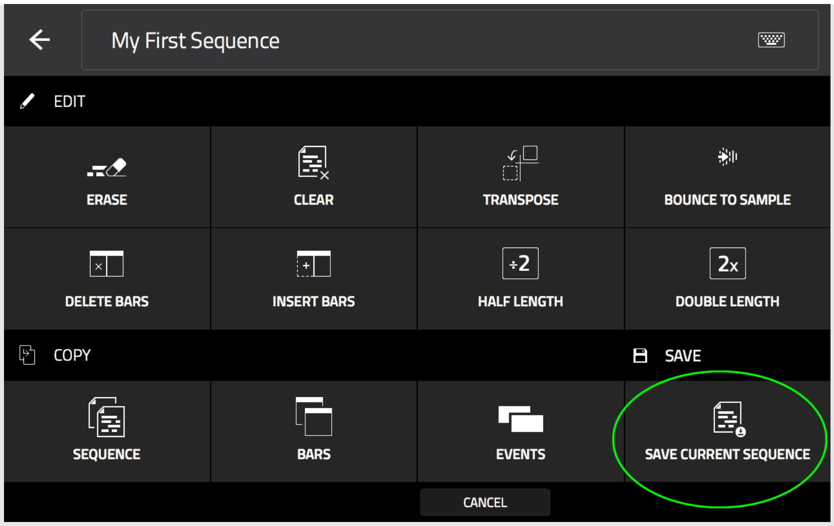
When you are finished playing, press **STOP**. To hear what you just recorded, press **PLAY START**. Congratulations, you just recorded your first MPC Software sequence!

Now, because of the way we set up the sequence ('**Timing Correct: OFF**'), your MPC will have recorded your performance *exactly* as it was played, including any mistakes! Don't worry, with a sequence nothing is set in stone and we're going to look at the many ways in which we can fix mistakes in the next tutorial.

If you've not set up autosave, remember to save your work using the project saving methods discussed in chapter 1.

Saving Individual Sequences

You can also save an individual sequence. In the **SEQUENCE** row in **MAIN**, press the **pencil icon**:



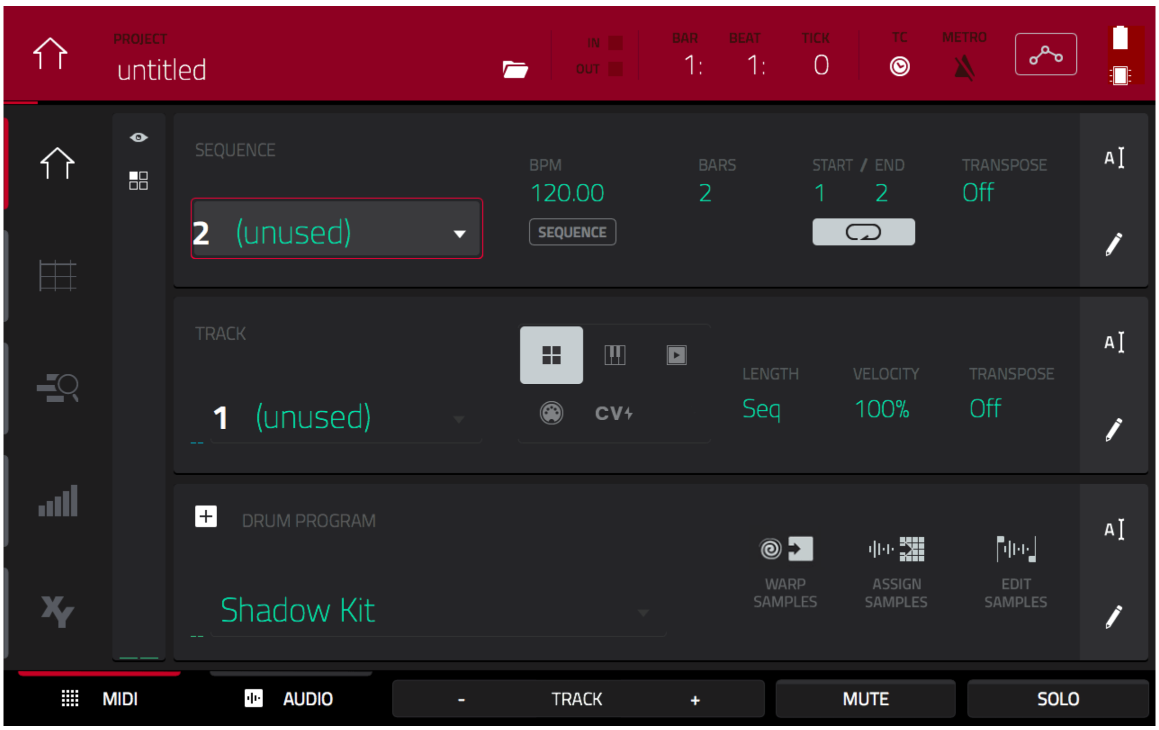
Select **SAVE CURRENT SEQUENCE**, choose a save location and hit **SAVE**. This will create an ‘**SQX**’ sequence file, which can be loaded into any project by any of the standard file loading methods we’ve used for sounds and programs.

Just remember that a sequence file only stores sequence event data - by itself it contains no sounds or programs, so unless you have a specific reason to save the sequence file separately just save the entire project and your sequence will be saved along with every other resource and setting in your project.

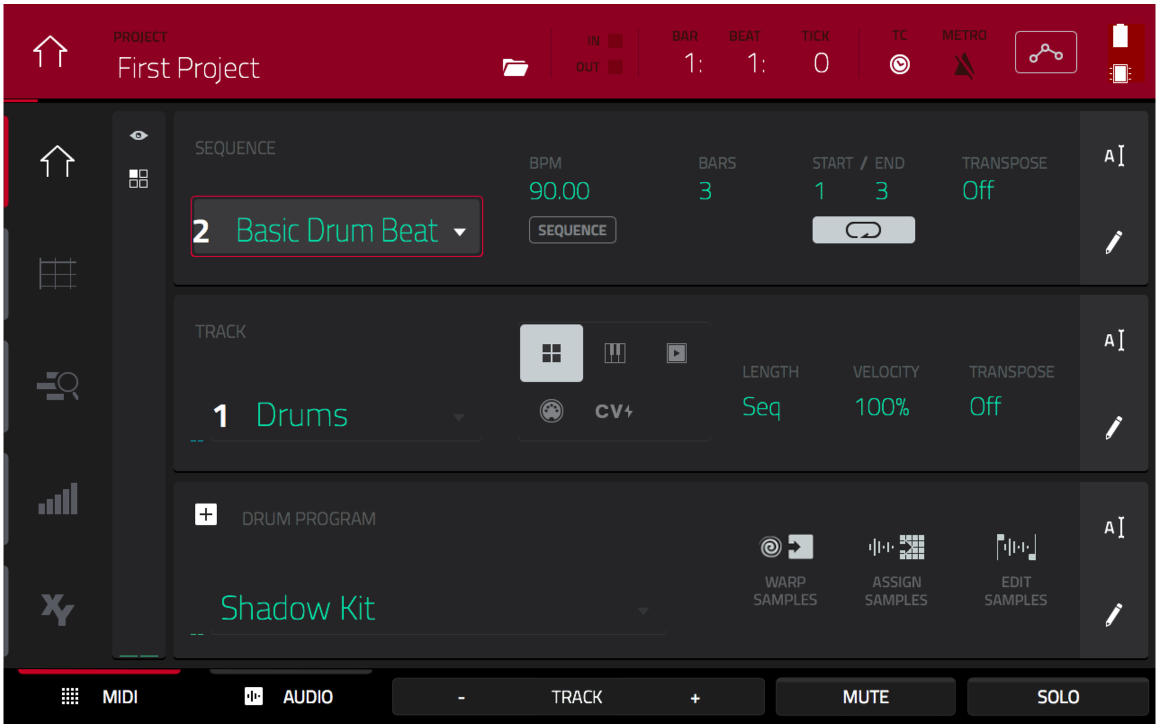
004 First Look: Grid Edit Mode

The 'Grid' is the MPCs visual representation of your current sequencer track allowing you to easily manipulate MIDI event data via the touchscreen interface.

Let's load up my own simple drum beat. In **MAIN** tap on the current sequence name (**1**. **My First Sequence**) and turn the data wheel (or click the + button).



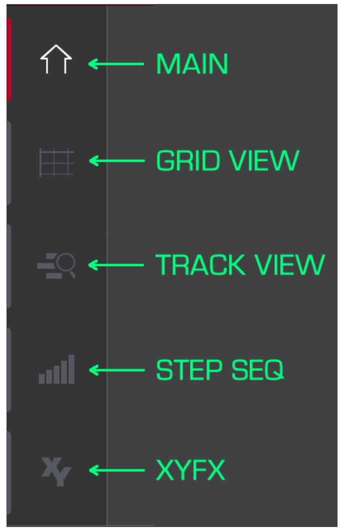
You've now selected the next sequence in our project (sequence 2), which as you can see is currently 'unused' (blank). Go to **BROWSER** and in the **004** folder load up the **Basic Drum Beat.sqx** sequence file by double tapping it – it will automatically get assigned to the selected blank sequence 2.



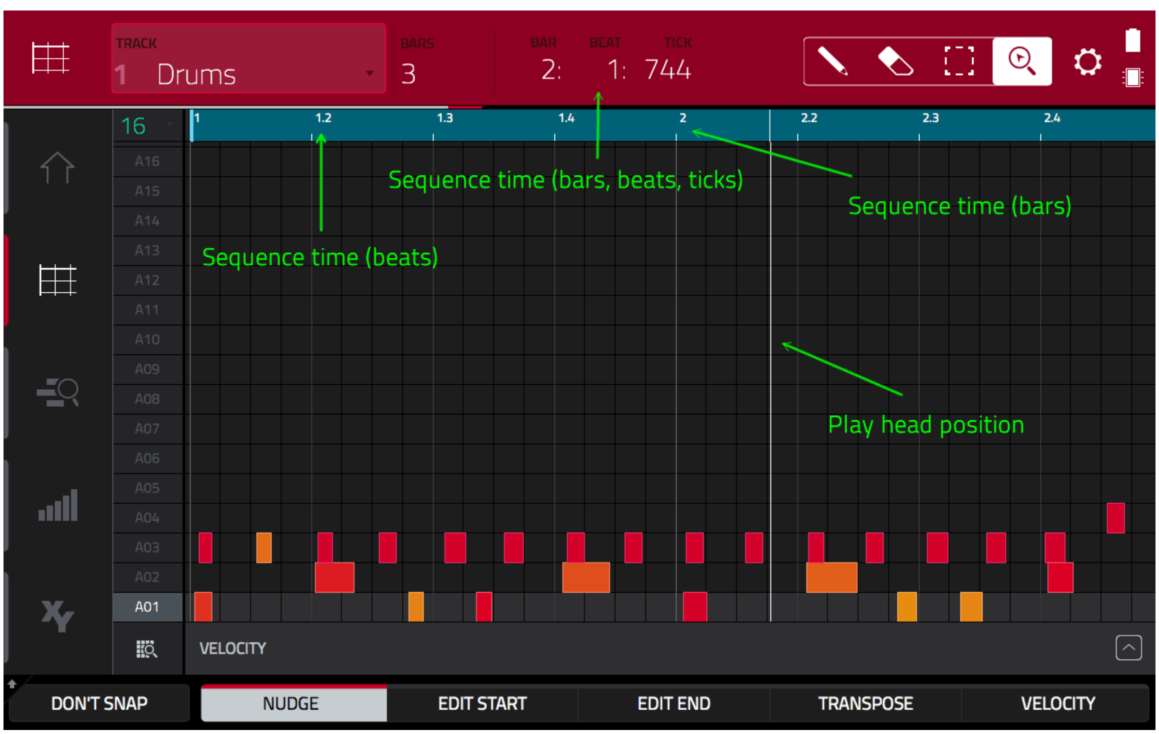
This is a drum sequence I recorded myself using the **Shadow Kit** we used in the previous chapter. Press **PLAY START** to listen to the drum sequence – it's a short break that I played 'finger style' with FULL LEVEL 'off' and no timing correct (TC/Timing Correct: OFF). It's a bit sloppy in places, but we're going to use the MPC's sequence editing features to make some adjustments and also add some more 'flavour' to the beat.

Understanding the Sequencer Grid

Let's now view all the recorded events from this drum track in **GRID VIEW**. To access the grid, you can go to **MENU > GRID VIEW**, or you can access it directly from MAIN by using the left hand side '**GRID VIEW**' shortcut:



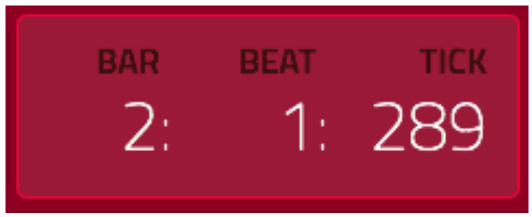
The sequencer grid provides us with a visual overview of our sequences as well as allowing us to easily manipulate our sequence to change any aspect we wish. Across the top of the screen is the **sequence time line**:



This indicates the time location of sequence events and the play head. Sequence time in MIDI is not normally measured in minutes and seconds, instead it is measured in **bars**, **beats** and **ticks**.

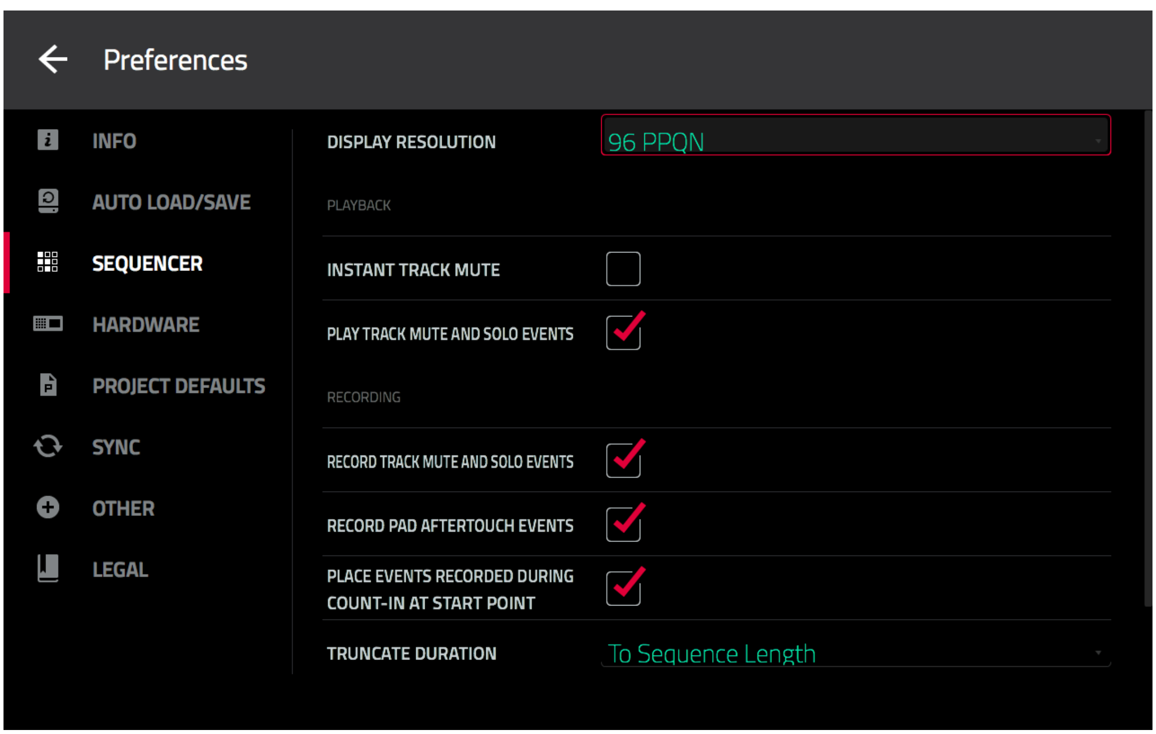
The current 'Beats & Bars' time is shown in blue at the top of the grid timeline, where, for example, '**1.3**' represents '**bar 1, beat 3**'.

For a more accurate representation of the time the play head is currently set at, you can refer to the sequence time indicator at the top of the page:



In the graphic above, our current sequence time (i.e. the location of the sequence play head), is located in **Bar 2**, **Beat 1, Tick 289**. **'Ticks'** are the smaller increments between each beat (a maximum of 960).

If the 'ticks' are only showing values two digits long, head over to **Menu > Preferences > SEQUENCER > DISPLAY RESOLUTION**:



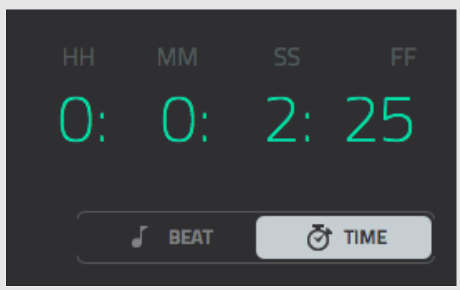
The display resolution of the MPC Software can be set to **960 PPQN** or

96 PPQN. So consider a sequencer event recorded at **1:4:723**. If your display resolution is 96 PPQN this time will be displayed on screen as **1:4:72**. If your display resolution is 960 PPQN then the time is displayed as **1:4:723**. In both examples, the *actual* time location stored internally is always 1:4:723. So this option is really nothing more than a way of rounding off and simplifying the sequencer display time (it makes it look like the time display in a legacy MPC, although older MPCs).

I feel this is a bit of a pointless option, so throughout the book I will assume you have set your display resolution to **960 PPQN**:

Sequence Time in Hours/Mins/Secs?

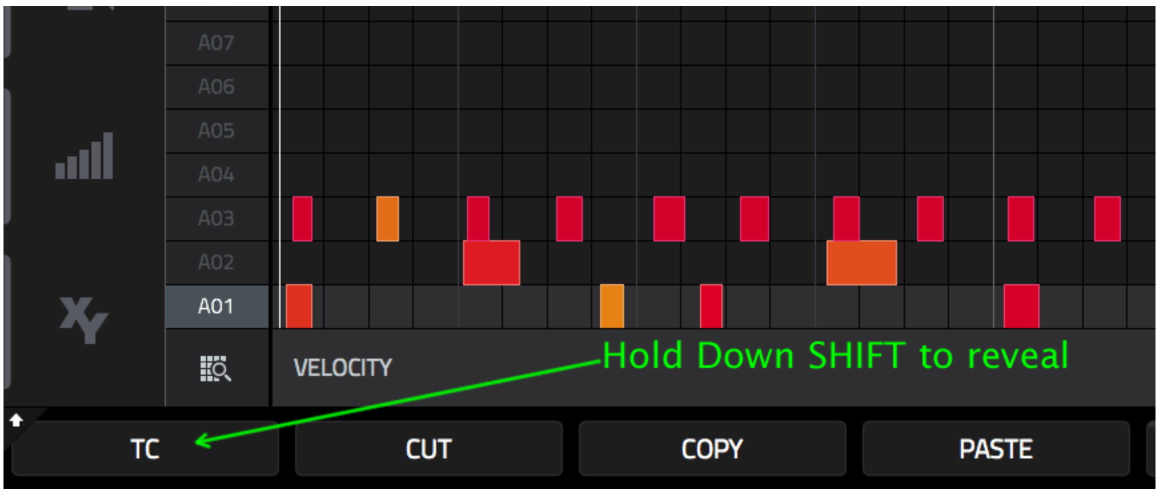
Double tap the sequence time indicator to display the **LOCATE** window, which also provides an option to switch to more traditional time display of hours, minutes and seconds.



While this has no real use in terms of MIDI events, it can be helpful when syncing your MPC with DAWs or other audio based recording devices.

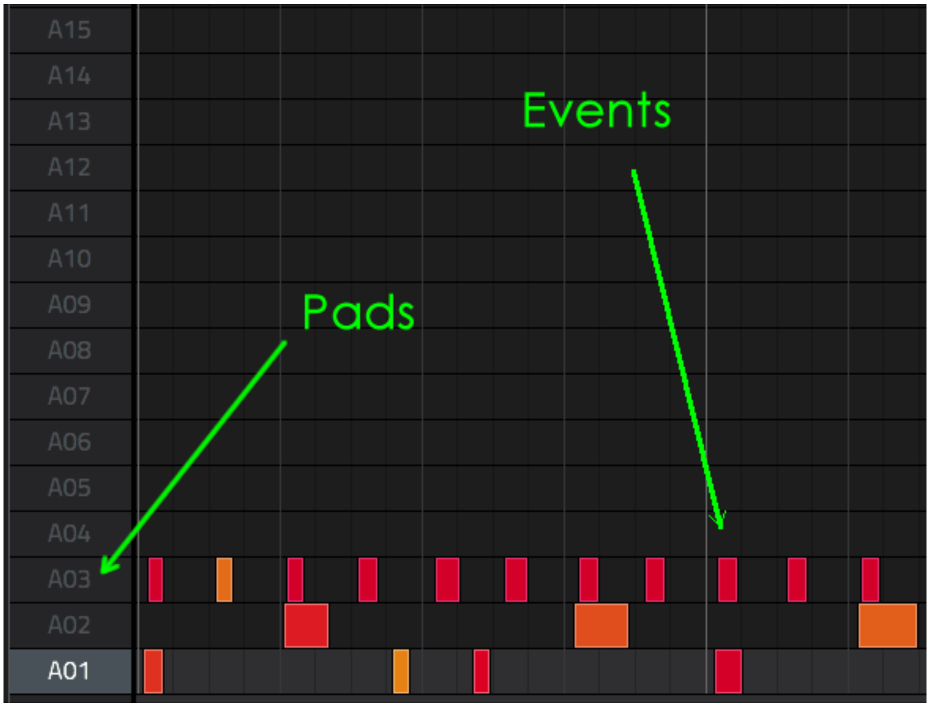
In the top left of the GRID is an option to set the '**Timing Correct**' (which we briefly met in chapter 3).

If this isn't set to OFF, double tap and select '**OFF**' from the pop up list. Alternatively you can set TC from the TIMING CORRECT window as we did in the last chapter; in GRID MODE the **TC** window is accessed by holding down **SHIFT** and pressing the **TC** button at the bottom left of the page.

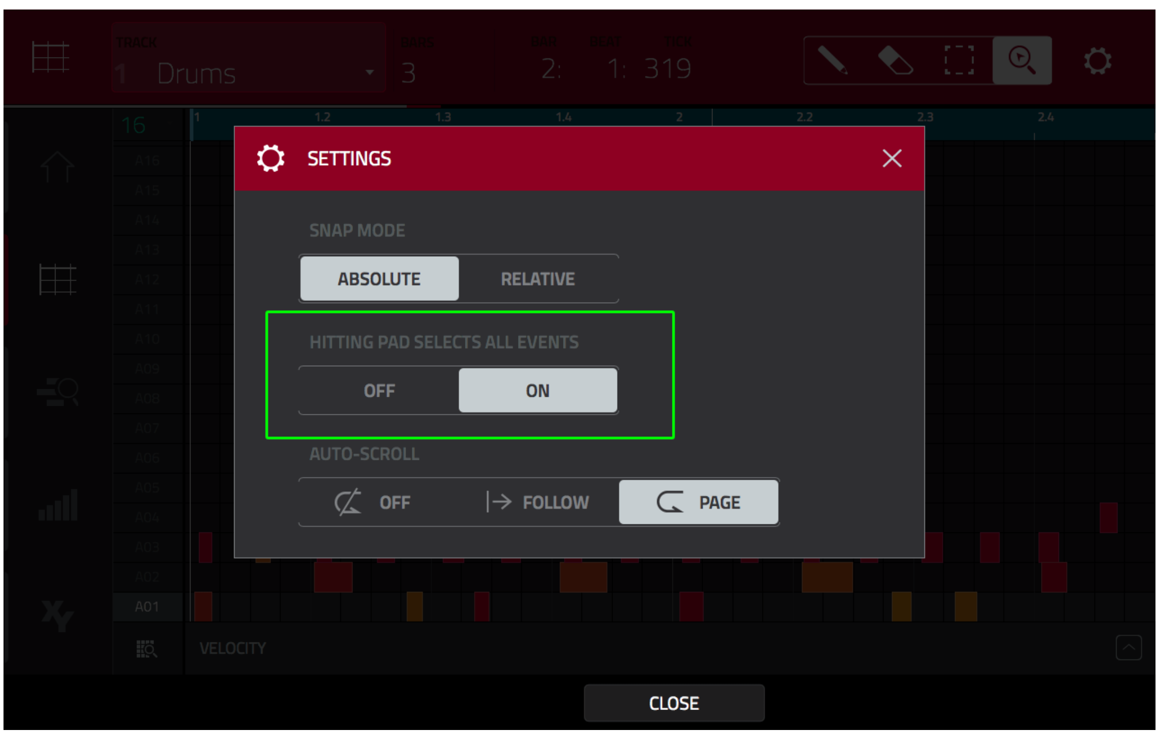


Later in this chapter we'll start working with some of the alternate settings, which can be used to help fix timing errors.

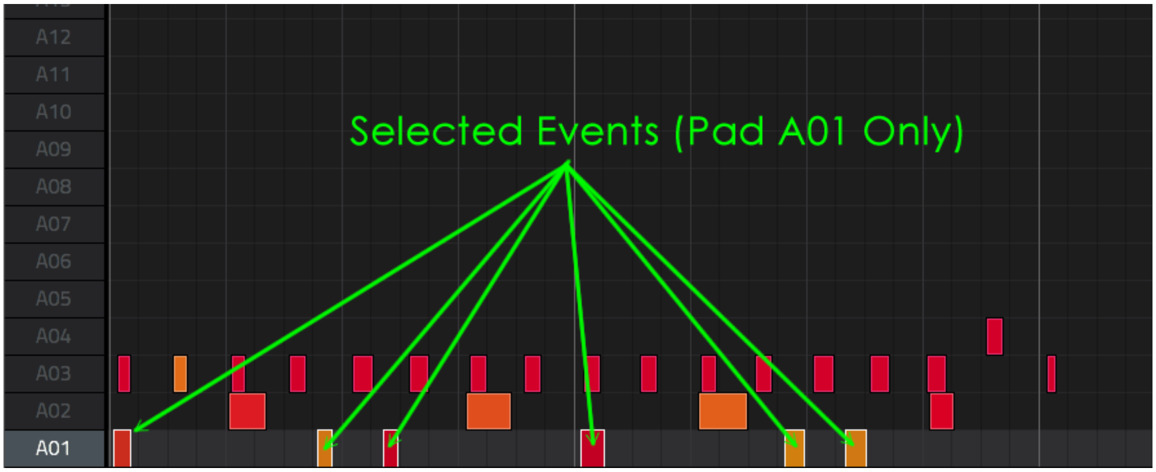
The main portion of the grid shows the currently recorded **events** as a series of coloured rectangles; the longer the rectangle, the longer the duration of the event. The more red, the higher the velocity of the event; i.e. a fully red event represents a hard pad hit, a yellow event is a very soft hit.



Down the left hand side of the grid are the pad number indicators – this shows you which pad each line of events relate to. Press the top right '**SETTINGS**' 'gear' icon:



Makes sure '**HITTING PAD SELECTS ALL EVENTS**' is set to **ON**. Now touch the **A01** line in the pad column, or simply hit pad **A01** on your controller:

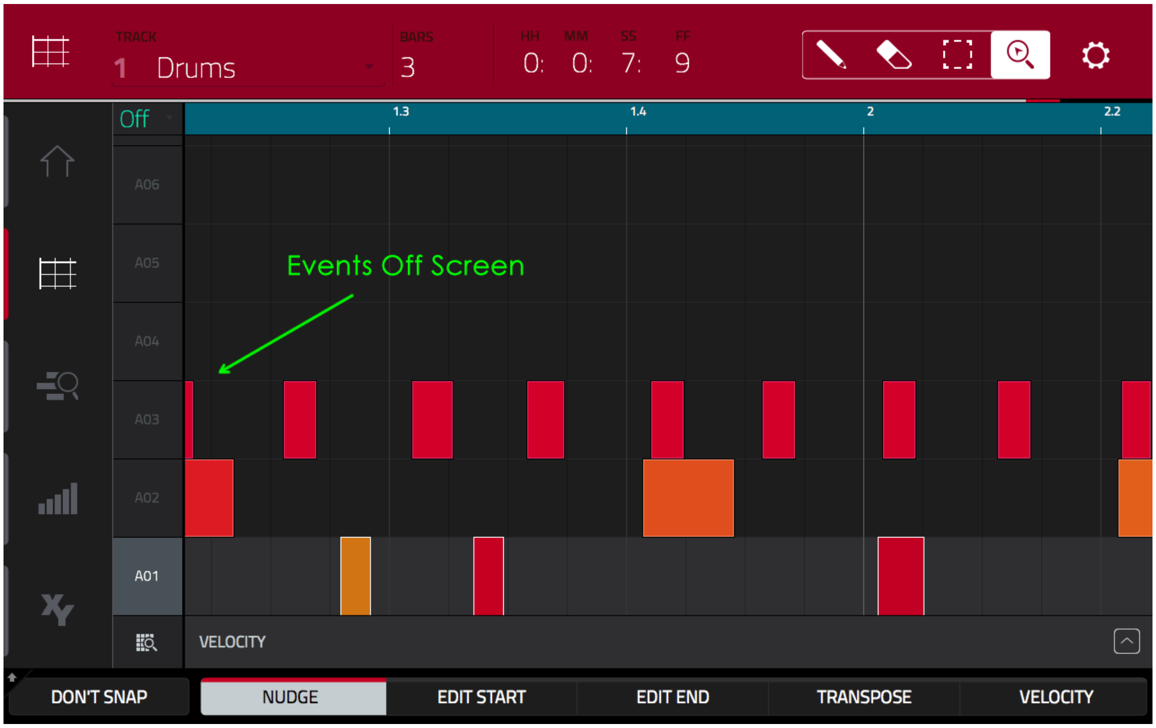


This selects all the pad A01 events in the grid – as pad A01 is our kick drum sample, this row represents the kicks in our beat. Row **A02** represents the **snares** (pad **A02**), row **A03** are all our closed hats on pad **A03**, and row **A04** represents any **open hats** from pad **A04**.

To magnify the GRID first make sure the **magnifying glass** icon is selected in the top bar of the screen:

To magnify the grid events, use a 'pinch and zoom' technique; place your index finger and thumb together on the screen and pull them apart while maintaining light downward pressure of on the screen. As you do, the screen will magnify at the position where you are pinching. Apply this horizontally to perform a horizontal zoom, vertically for a vertical zoom.

At this point many events will now disappear off screen:



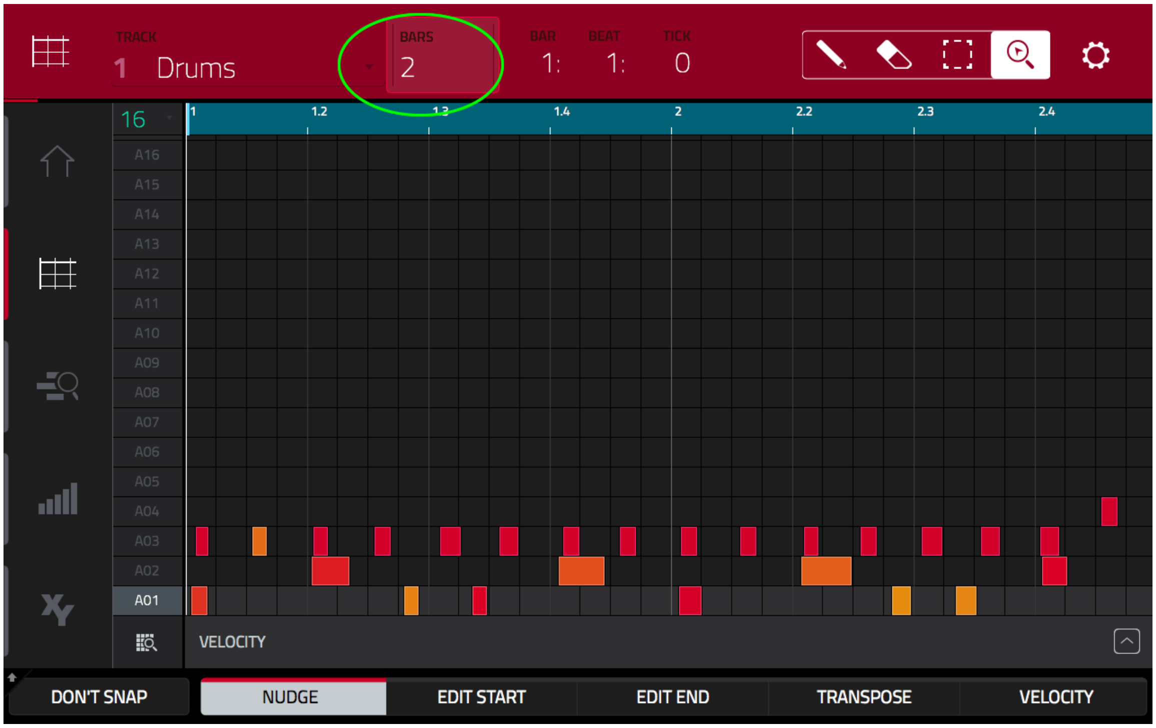
To view those events, simply 'drag' the screen with the tip of your finger. In the example above, to reveal the events to the left hand side of the screen, drag the screen to the right. To quickly return to the default screen magnitude, press the **zoom reset** icon:

'Screen dragging' only works if the zoom tool is selected.

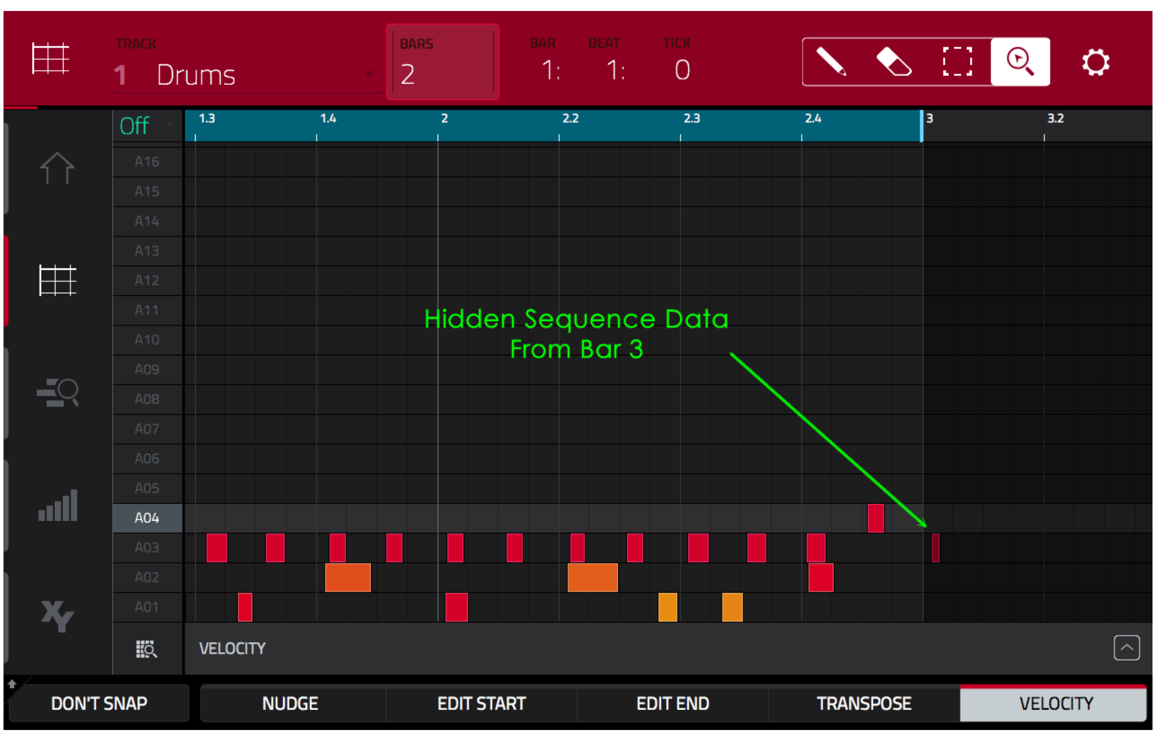
Basic Sequence Editing

Currently our sequence is 3 bars long due to the additional closed hat that was recorded at the start of the third bar. We can however easily turn this sequence into a tidy 2 bar drum loop.

Single tap the '**BARS**' parameter at the top of the screen using your cursor and turn the data wheel anti clockwise so it reads **2**.



Now the MPC will currently only play the first two bars of this sequence. With the zoom tool selected, pinch and zoom 'inwards' to reduce the magnification so you can see that the event data within bar 3 – it's still there, but it's now darkened as it's currently not being used:



If in the future you were to extend the number of bars (to 3 or greater), this data would once again become available for playback.

Now briefly head over to **MAIN** and set '**Loop**' '**ON**' as we did previously (there's no way to set this from the GRID page). Hit **PLAY START** and your sequence now continuously loops every two bars until you press the **STOP** transport button.

Overdubbing

How about we add some more drums – perhaps add an open hat in the first bar? One option is to **overdub** to the existing track. Overdubbing is a little different to straight recording as the overdubbing process *retains* the existing note data on a track while simply merging your new performance on top of it.

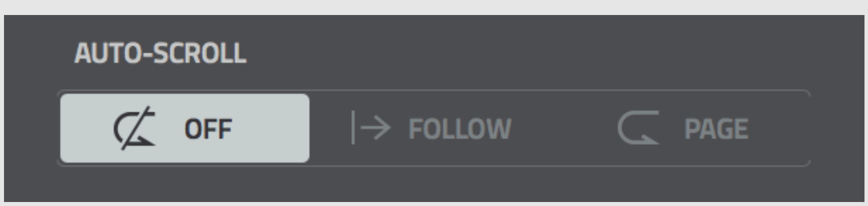
First 'arm' the MPC into overdub mode by pressing the dedicated **OVER DUB** transport button. When you are ready to record, hit **PLAY START**.

At this point your count in starts as normal and then the MPC begins the overdubbing process. As no data is being overwritten, you can happily leave the MPC overdubbing in a continual loop until you are happy to lay your sounds down. Once you've added your open hat, press **STOP** and hit **PLAY START** to have a listen.

Not happy with what you recorded? Well you have a few options, but the easiest is to just hit the **UNDO** button and this will 'undo' the entire overdub you just made. You can then try the overdubbing process again. If you change your mind, simply hit **SHIFT** and **UNDO** together and this will perform a '**REDO'** – i.e. it will put those erased events back (basically an undo of an undo!).

Controlling the GRID view during playback

If your current track has events that are 'off screen' (e.g. because you are zoomed in, or you simply have a long sequence), you can control the way the MPC displays these events. Hit the **SETTINGS** icon in the top red bar:



With **AUTO-SCROLL** '**OFF'**, the GRID will remain completely static during sequence play back. With it set to **FOLLOW**, the GRID will gradually move and follow the play head in real time. With it set to **PAGE**, the GRID will stay static until the play head reaches the edge of the screen, at which point the entire screen moves to the next 'page' of events. I generally use the '**FOLLOW'** settings.

Load up the sequence '**Overdub.sxq'** from the **004** folder. Go to **MAIN** to select this new sequence (it should be loaded as sequence 3). This is my version of the sequence so far, with an additional open hat overdubbed just before the start of bar 2.

Recording With Looped Sequences

If you have a looped sequence (**LOOP:ON**) and record to a track using REC, the sequence will revert to an **OVER DUB** after the first loop. This way you avoid accidentally wiping over any data you recorded during the initial play through of the sequence. This also means you can listen back and over dub in the same recording session.

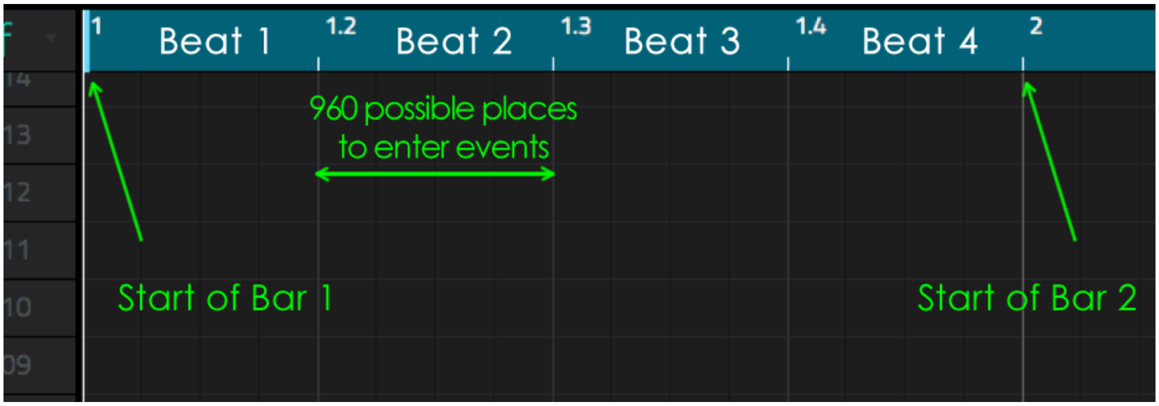
Understanding Sequence Resolution

Before we look at moving events, it's important to first understand how 'time' is handled in a MIDI sequence, because 'moving' an event is simply another way of saying 'changing the time' the event is triggered.

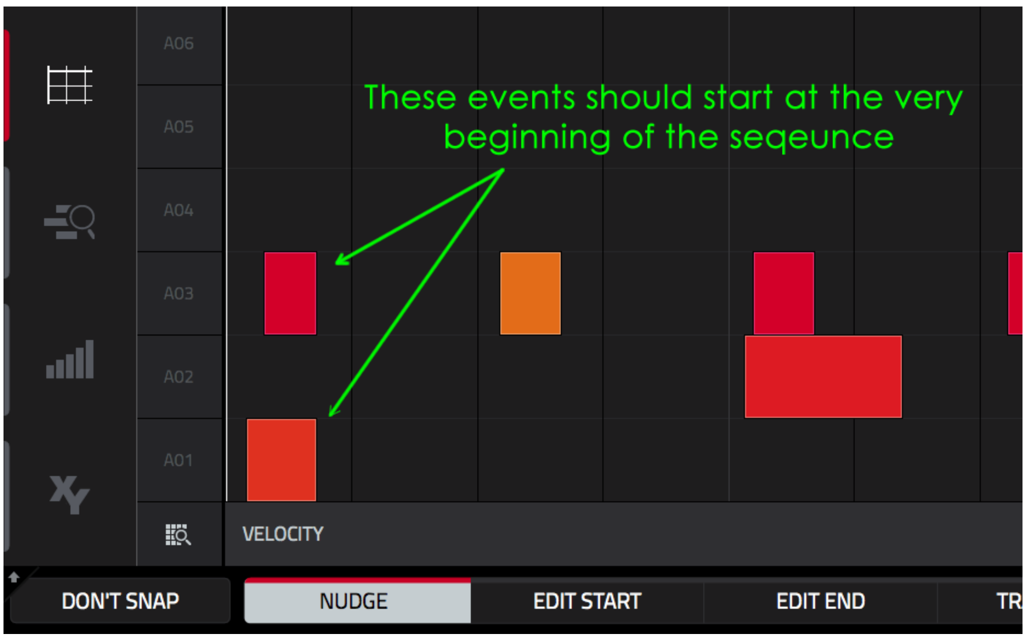
I've already mentioned that sequencer time is measured in 'bars, beats and ticks', but what exactly is a 'tick'?

The sequencer in an MPC is said to have a '**resolution'** of 960 Parts Per Quarter Note **(960 PPQN**). In a sequence with a time signature of 4/4, a 'note' is a bar (made up of 4 beats), so a 'quarter note' is simply 1 beat.

This means each beat in your sequence contains up to 960 possible places where an event can be inserted. Most legacy MPCs only have a sequencer resolution of 96 PPQN, so the MPC Software has ten times that resolution, which effectively allows us to record event performances incredibly accurately.

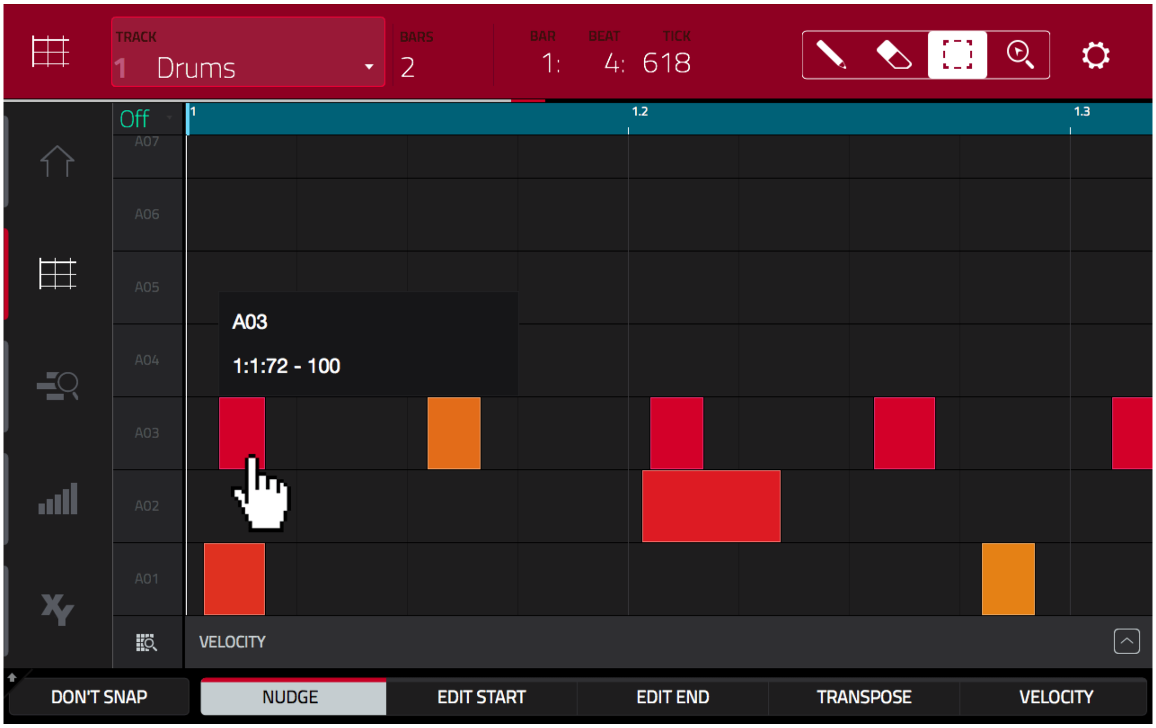


Let's take a look at our existing sequence. Firstly pinch and zoom in so the events are nicely magnified and drag the grid so we can see the first few events.



Next, we'll need to choose the '**select**' tool from the top red bar:

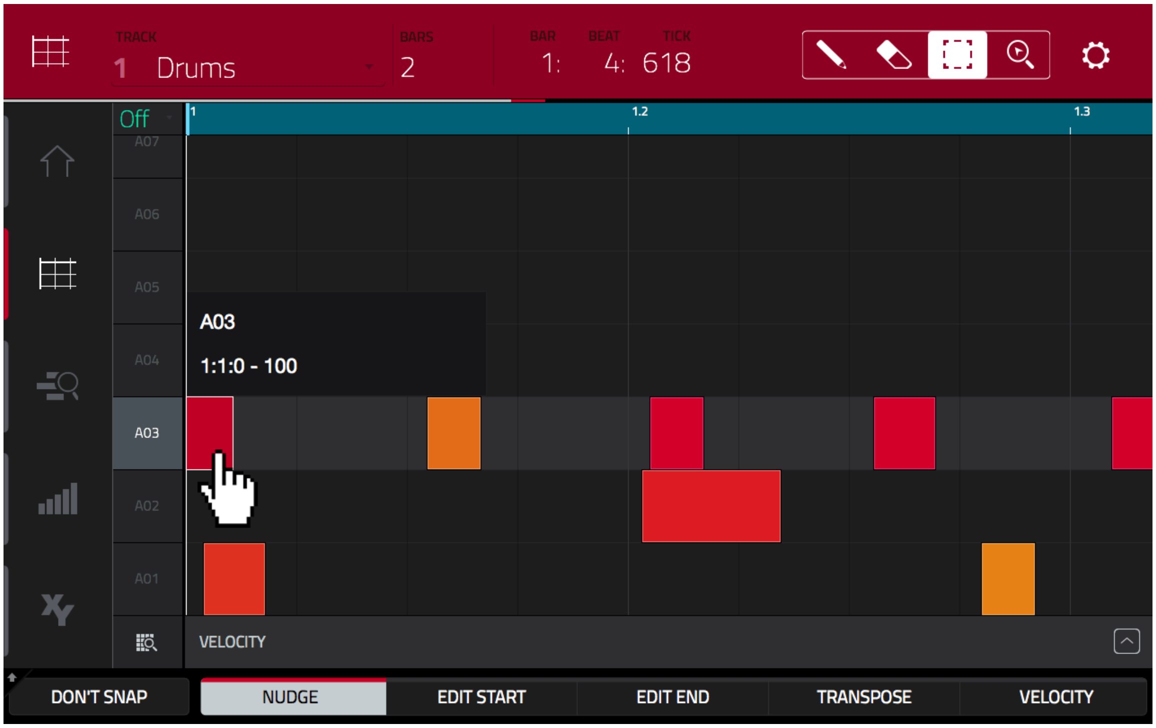
At the start of the sequence, press the first red event in the **A03** line. Firstly, you'll hear the (hi hat) event play back, and you'll also see this:



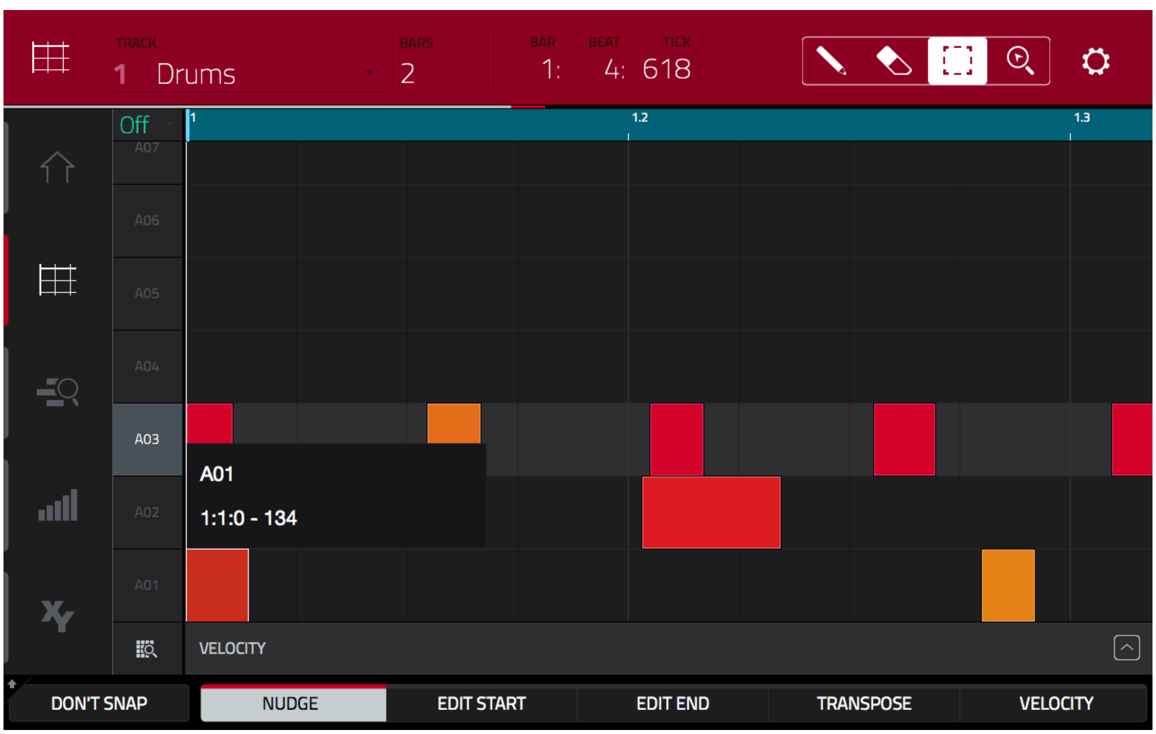
The MPC is now showing you some basic information about this event.

- **A03** – this is the pad
- **1:1:72** – this is the time position of this event in the sequence in 'bar, beat, tick' format. So this event is positioned in bar **1**, beat **1**, tick **72**.
- **100** – this is the length (duration) of the event.

This hi hat was actually supposed to play at the very beginning of our sequence, nice and tight in position **1:1:0** (i.e. bar 1, beat 1, tick 0), so let's manually move it there. There's two ways to move an event; firstly, you can just 'drag' it over. As you drag the black pop up will indicate the sequence time you are moving it to:



For more accuracy we can alternatively use the data wheel or +/- buttons. Touch the first **A01** kick event so it is selected. Now press the **NUDGE** button at the bottom of the screen. With NUDGE selected, move the **A01** kick event to **1.01.0** by turning the **data wheel anti clockwise**, or press the **'-'** button several times.



Press **PLAY START** and you'll hear that the sequence begins much more 'tightly'.

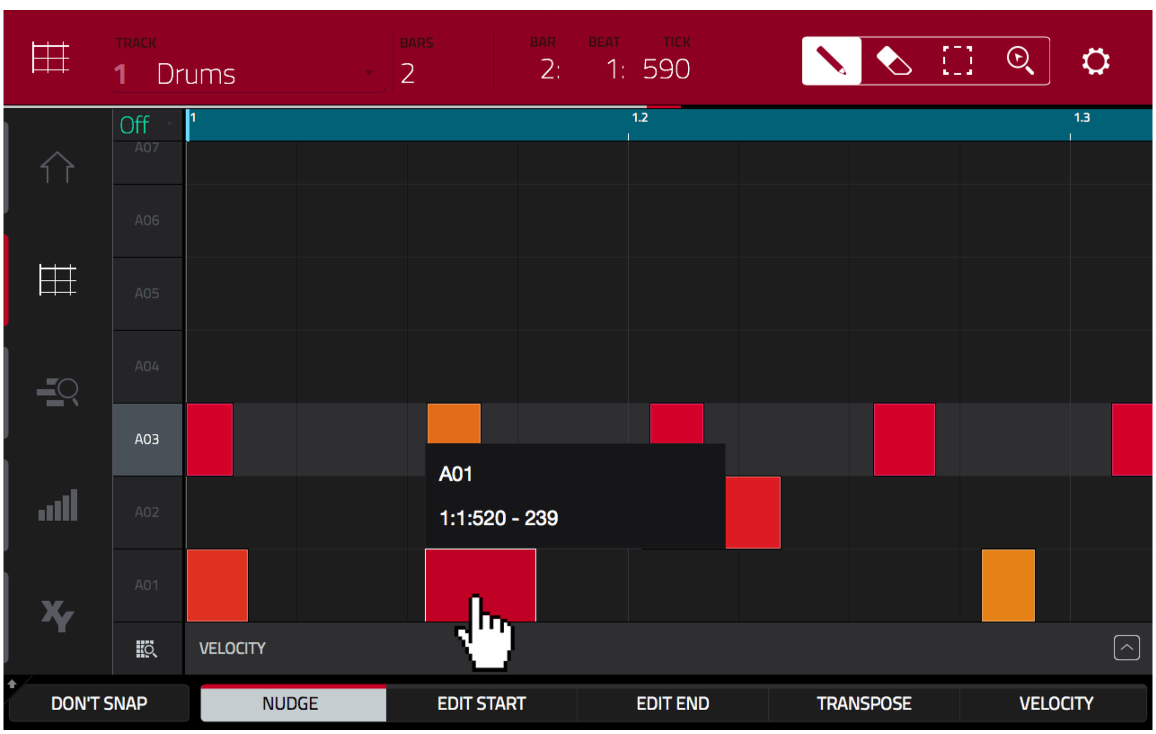
Adding New Events Manually

Select the '**Draw**' tool from the top right of the screen:

Let's use the draw tool to manually insert a kick event into our sequence – whenever the draw tool is selected we can add an event to our track simply by tapping the grid.

Tap the grid on the **A01** row, in line with the hi hat on the pad **A03** row. If the kick isn't lined up with the hat above it in the A03 row, simply move it using the techniques covered previously. As this is a more precise move, you might prefer to use the data wheel or +/- buttons.

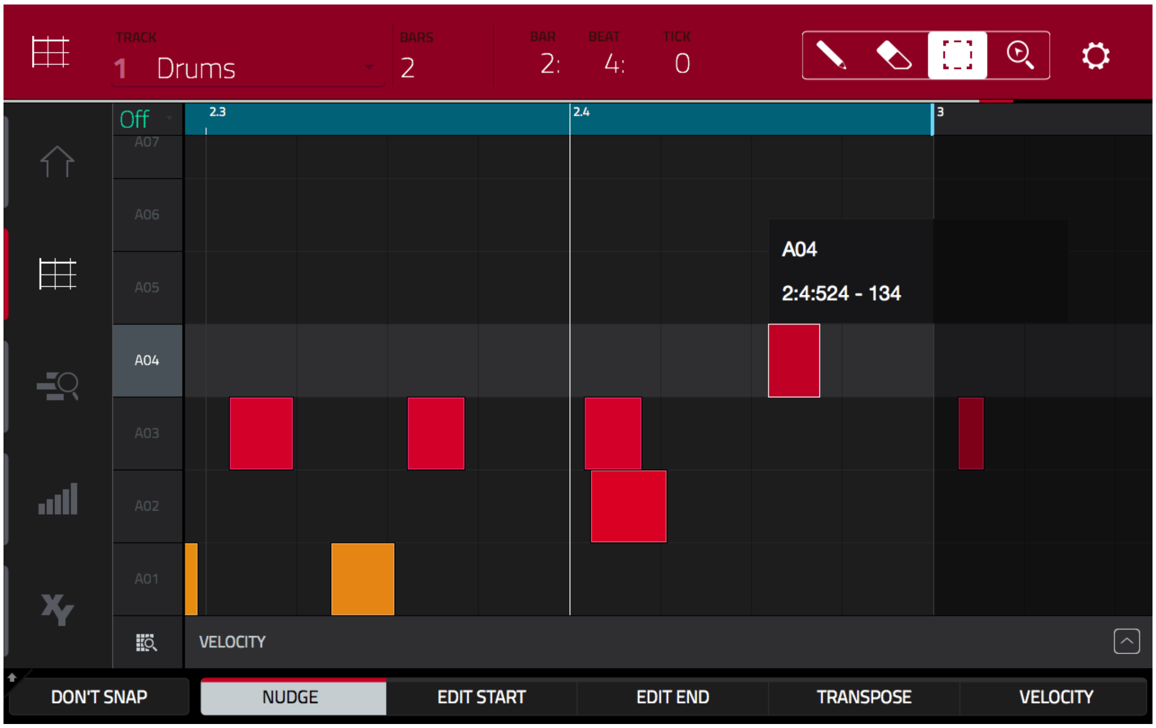
Move the event to **1:1:520** as shown below:



Press **PLAY START** to hear the sequence.

Transposing Pad Events

Hit the zoom tool and drag the grid to the left until you reach the end of the sequence - there you'll an open hi hat (pad **A04**) at sequence time **2:4:524**:



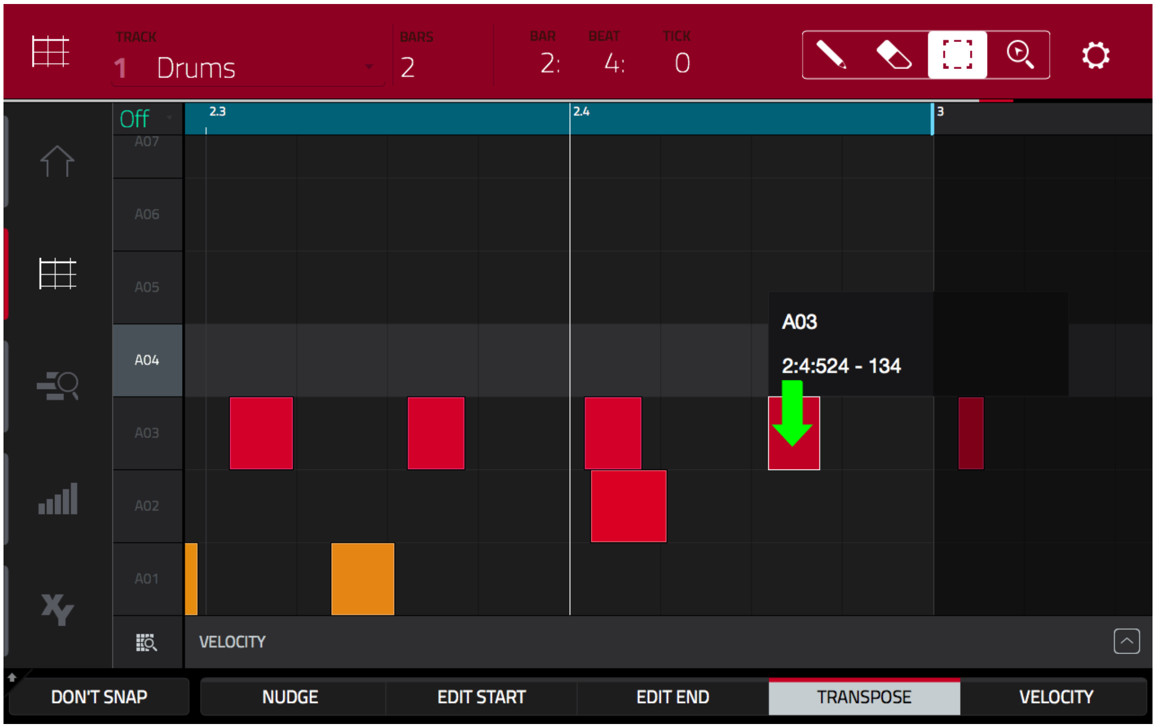
Let's change this event to a *closed hat* instead. Now, the closed hat in our kit is on pad A03, so one way to change the pad triggered is to simply move the event down to the **pad A03 row**.

Touch the '**Select**' tool and drag the event from the **A04 row** to the **A03 row**. Now, the problem with this method is you'll probably find is that when moving the event down it invariably also moves to the side, hence changing the event time.

The better method for changing the sound assigned to an event is to use **TRANSPOSE**. In musical terms, 'transpose' normally refers to changing the pitch of a note, but when dealing with DRUM events which are more often than not percussive in nature (they have no definable pitch), we use *transpose* to change the pad number assigned to an event. In this current case we want to change/ transpose the pad from A04 to A03.

Tap the **A04** event to select it and press the **TRANSPOSE** button at the bottom of the screen. Now, just like we did with NUDGE (to move events backwards or forwards), we can use the data wheel or +/- buttons to move our event, this time up or down so they trigger a different pad.

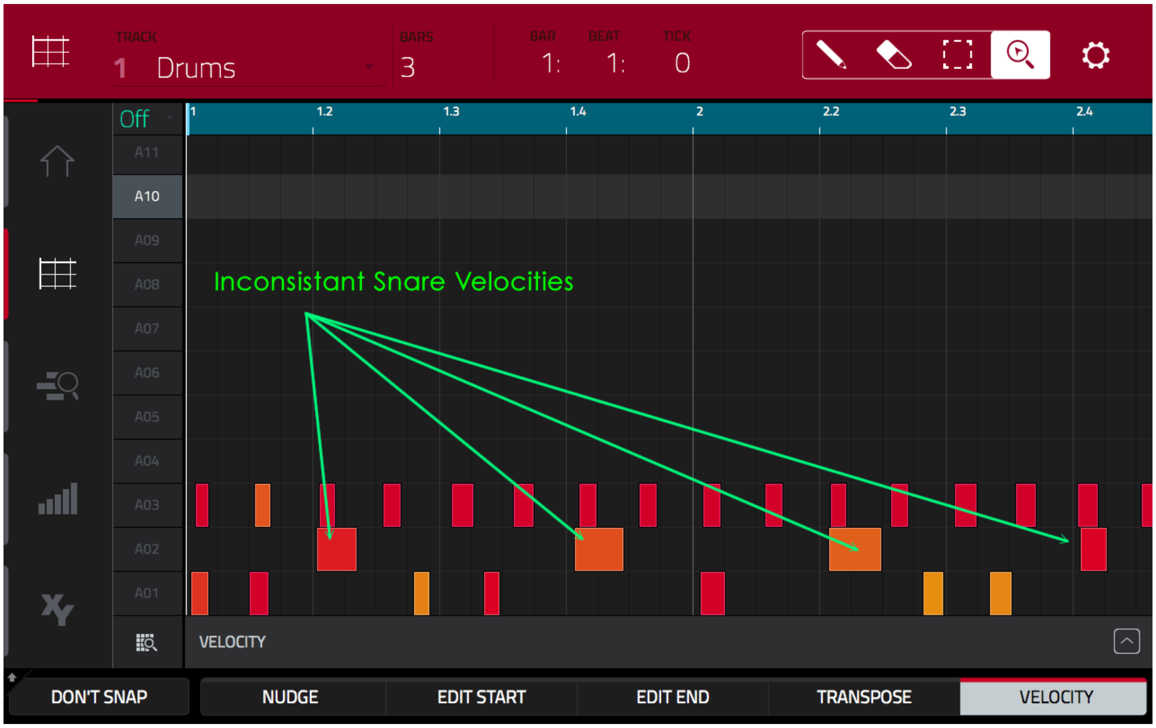
Press the '**-**' button to move the event down to pad **A03** and press **PLAY START** to listen.



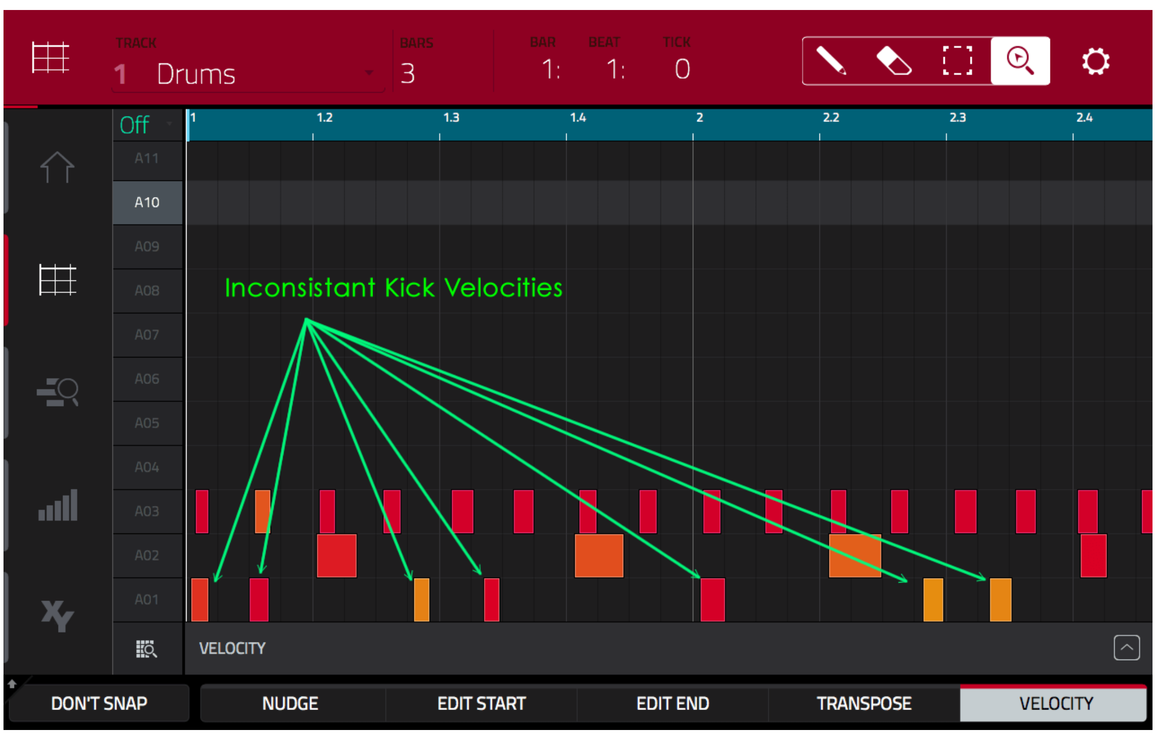
Editing Velocity

As this sequence was recorded with FULL LEVEL 'off', you should be able to hear quite a big variance in the loudness of many of the hits. Visually you can see this is the case as we have a range of event colours from yellow to red, with red being the loudest/hardest hits.

Take a look at the main snare hits on the 2nd and 4th beat of each bar, the difference in velocity on these hits is quite obvious with the last snare being quite red (hard), the third almost yellow (soft) and the other two snare events orange (medium hits).

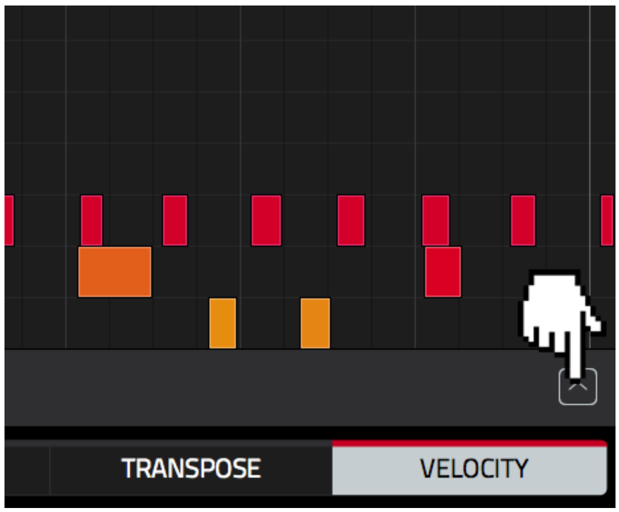


Similarly you'll see that the kicks on pad **A01** are somewhat inconsistent – you can also clearly hear this when playing back the sequence.

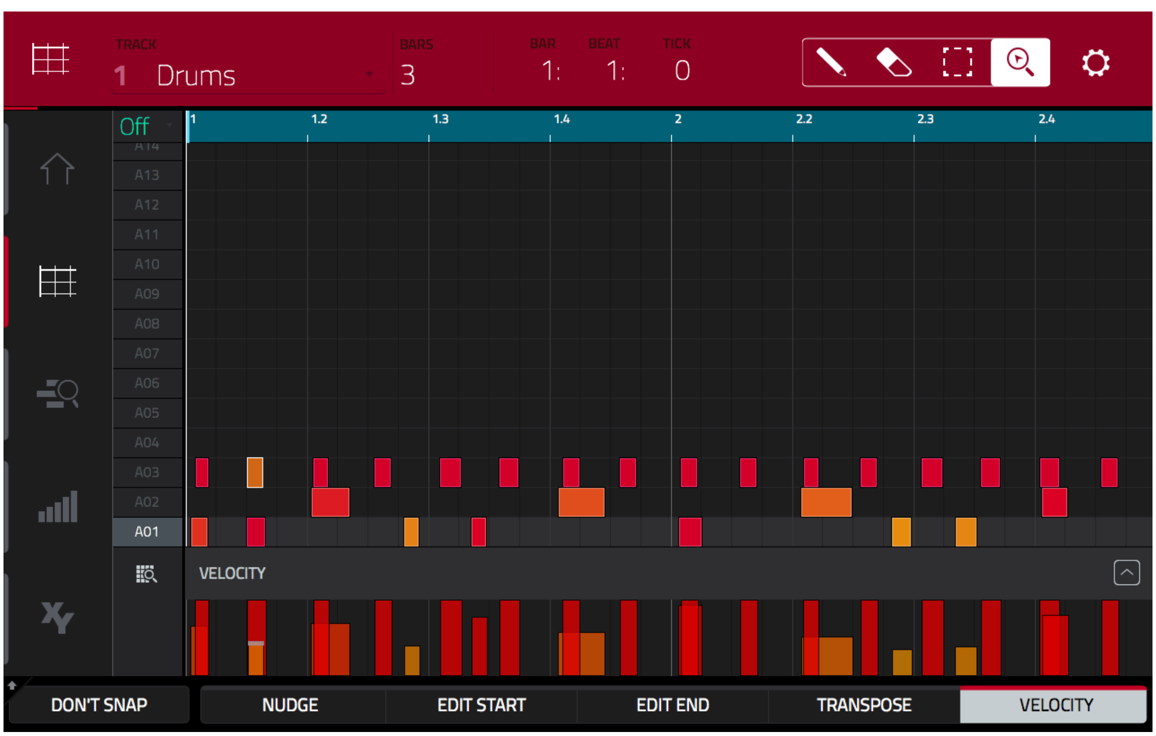


Now the whole point of recording with FULL LEVEL 'off' is that your sequences *do* have *some* dynamic variation as it helps give your performances some more natural qualities – if you don't want any dynamic variation just turn FULL LEVEL 'on'. However, the dynamics here are just too sloppy and erratic, but that's fine as it gives us an opportunity to earn how to manually edit velocities.

So, how do we edit velocities? Well, directly underneath the grid you'll see the **VELOCITY lane**; touch the up arrow at the end of the row to expand it:



You can now see a graphical representation of the **velocity** of each event:

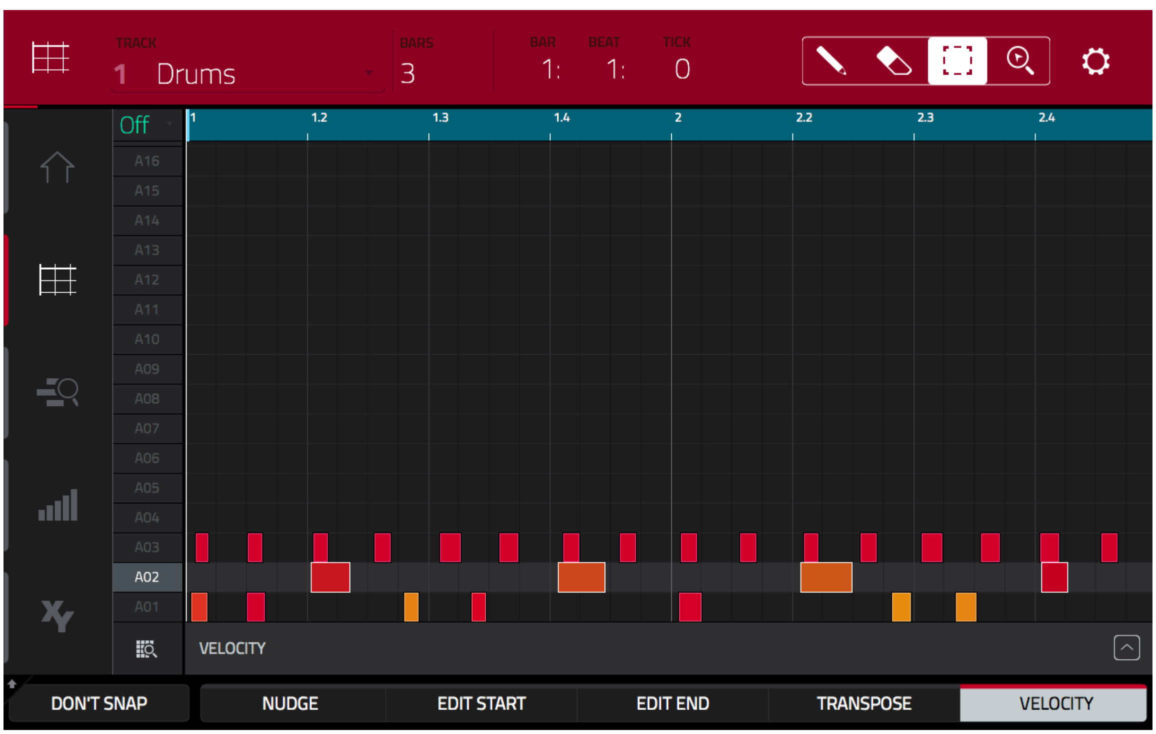


Magnify the start of the grid with the **zoom** tool so you have a close up of the second A03 hi hat event. Now choose the '**select**' tool and select it:

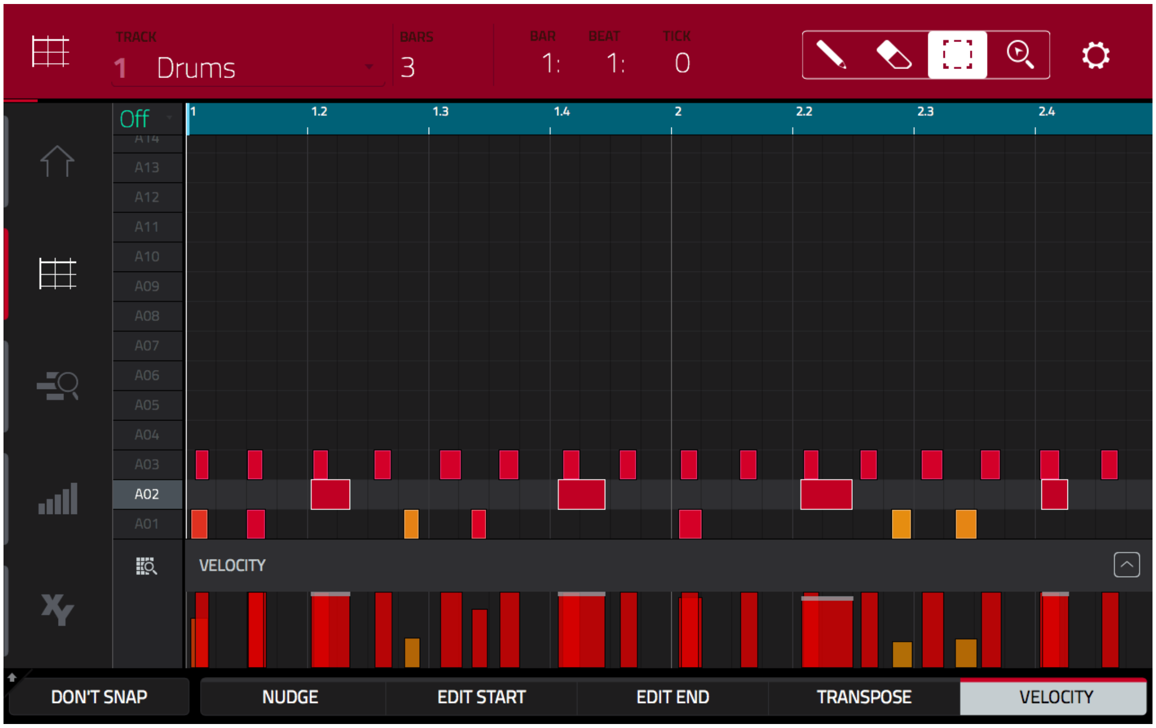
With the event selected, you'll see a grey line appear above its corresponding velocity:

Make sure the **VELOCITY** button at the bottom of the screen is pressed. With the velocity selected, start turning your data wheel clockwise to increase the velocity value – take it up so it is roughly the same level as the existing hats:

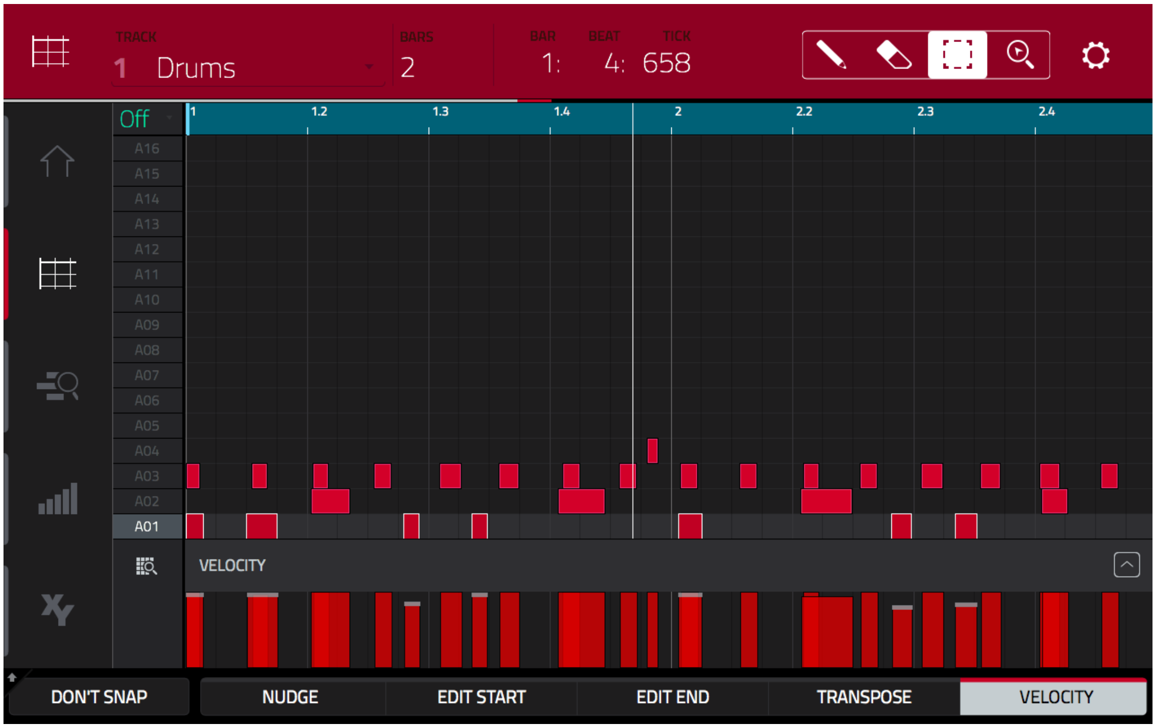
Go back to normal magnification using the zoom reset button. Let's get the snares to be the same, consistent velocity. As we've already seen, '**HITTING PAD SELECTS ALL EVENTS**' is set, we can quickly select all the snares in our sequence by hitting pad **A02**.



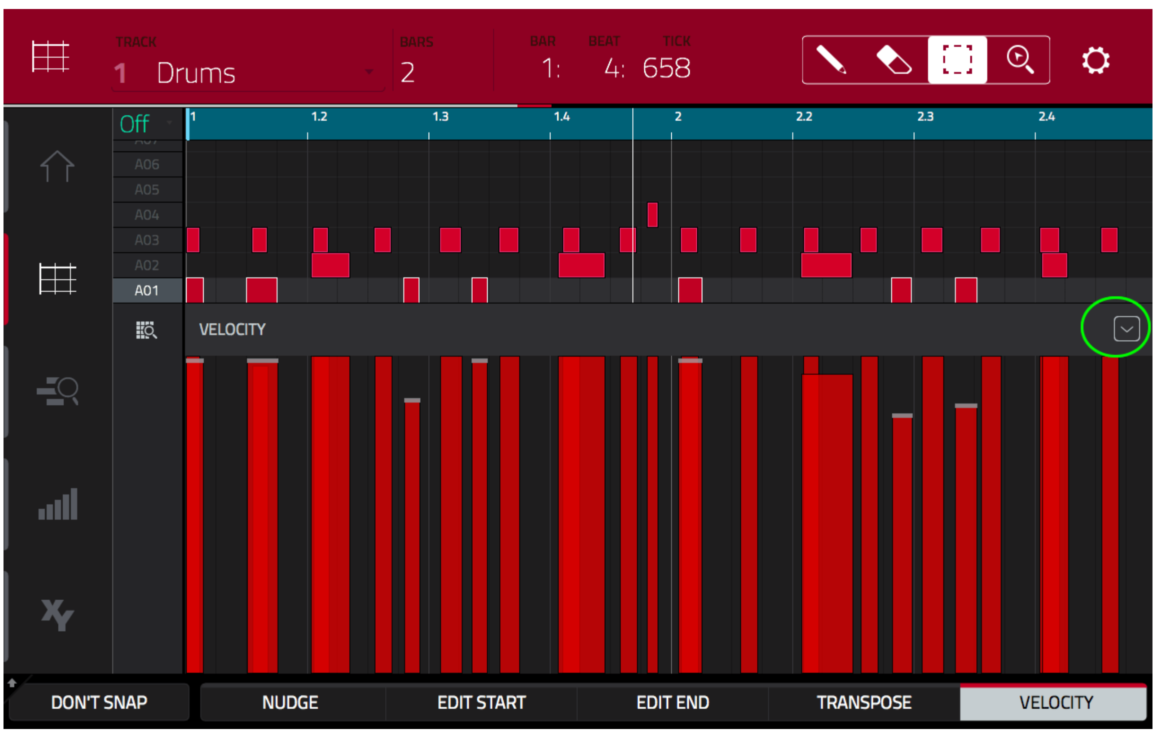
With all snares selected like this, any edit you perform will be applied to all the snares equally. Choose the '**select'** tool and 'drag' any of the existing velocity levels all the way to the top – all the other velocities will follow, leaving us with four snares at full velocity (doing it this way doesn't require pressing the **VELOCITY** button first):



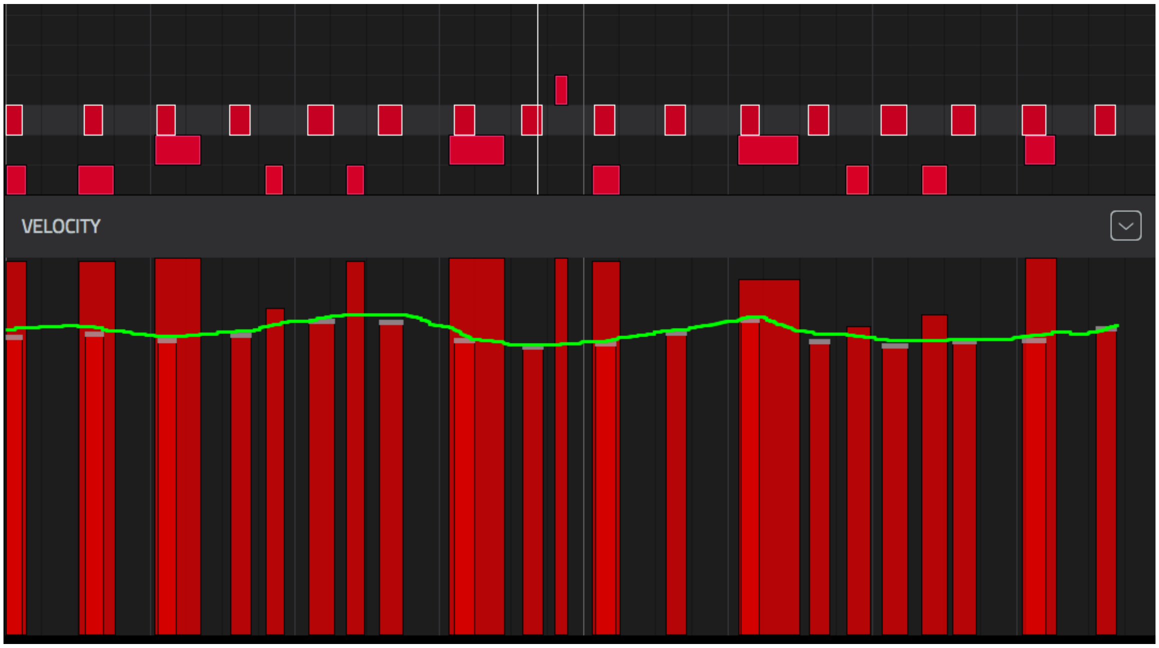
Repeat this process for the kicks, maintaining some minor variance between them:



One area where we can often benefit from more 'human feel' are hi hats. Let's improve the readability of our velocity lane – press the 'expand velocity' button once again to double the height of the velocity lane:



Select the **closed hat** (**A03**) row by hitting the **A03** pad. Now select the '**DRAW'** tool and starting from the first selected hat event, hold down and drag your finger across the velocity lane creating a slightly varying path – as you draw you'll see your hat velocities adjust themselves to your path (I've highlighted the path I took in green below) – I find drawing from right to left seems to work better:



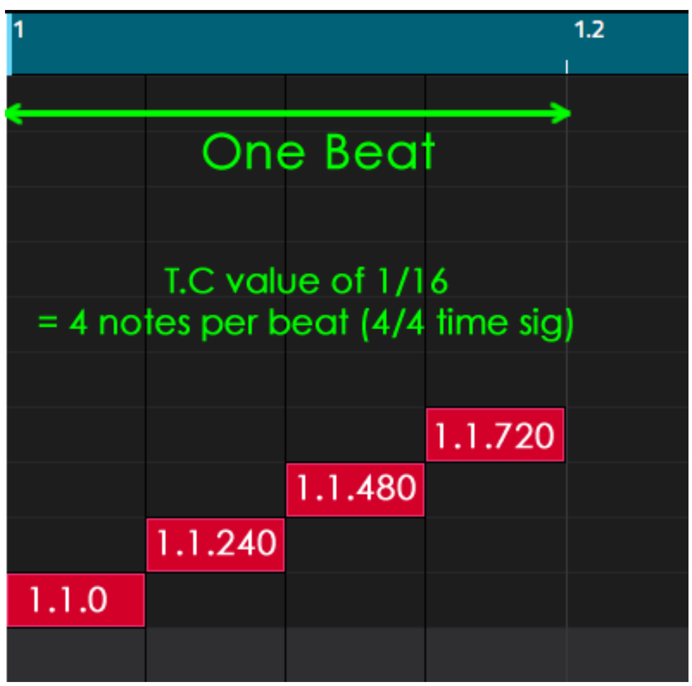
Press **PLAY START** to hear the improved version of your break. Load up my version – **Velocity Tweak.sxq**.

005 Quantising and Swing Essentials

We've discussed 'sequencer time' and how we have a resolution of 960 PPQN that can capture a performance pretty much as it was played. But sometimes we don't want the MPC to capture our performance exactly – if your timing isn't rock solid then your performances can sound sloppy, especially when it comes to drums.

To help compensate for this the MPC has a feature called **Timing Correct** (TC). Timing correct only allows events to be placed within specified points within a quantise grid. If an event is played outside an acceptable point in the grid, it is simply moved automatically to the nearest quantise point within the grid.

Take the example of a Timing Correct (T.C.) value of **1/16**. A T.C. value of 1/16 only allows 16 events to be placed in each bar, i.e. (in 4/4 music) four events per beat.



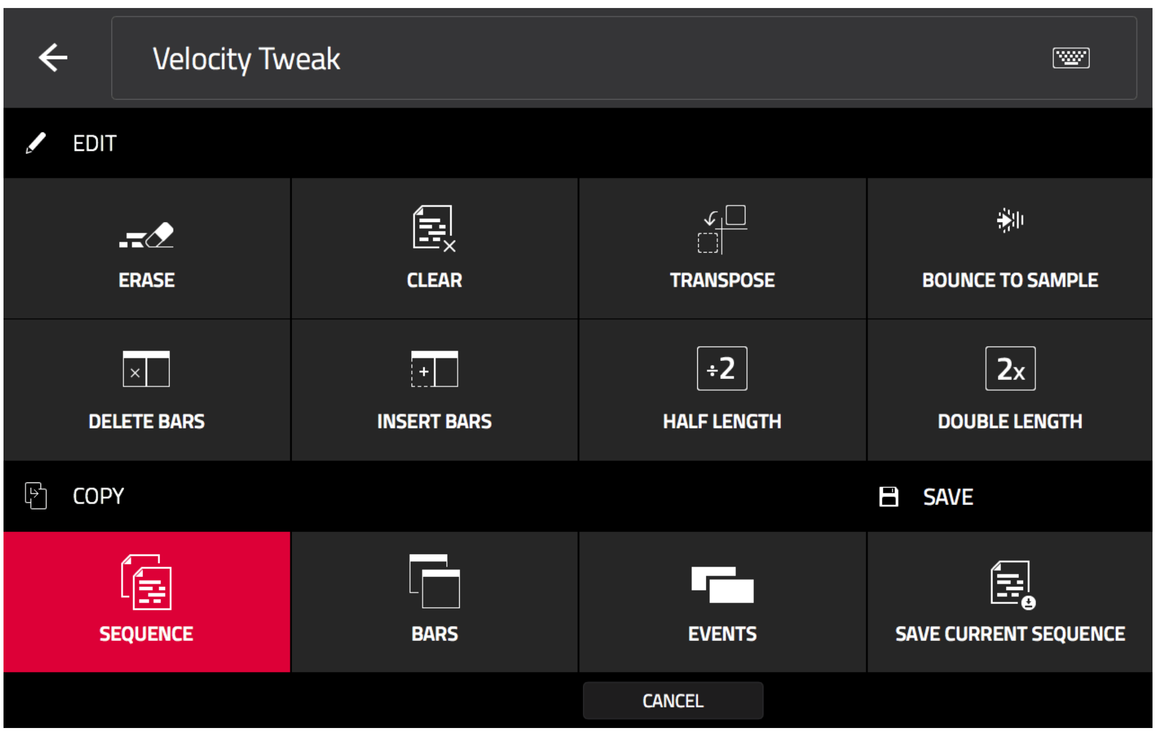
As indicated in the graphic above, when you record with a T.C. of 1/16, the MPC will only let you place events at ticks 0, 240, 480 and 720 of each bar. If you try to play an event at **1.1.187**, the MPC will just place that event at the *nearest* quantise point; **1.1.240**.

This can automatically 'fix' events that were played with poor timing, however it can lead to your music sounding somewhat rigid and almost robotic.

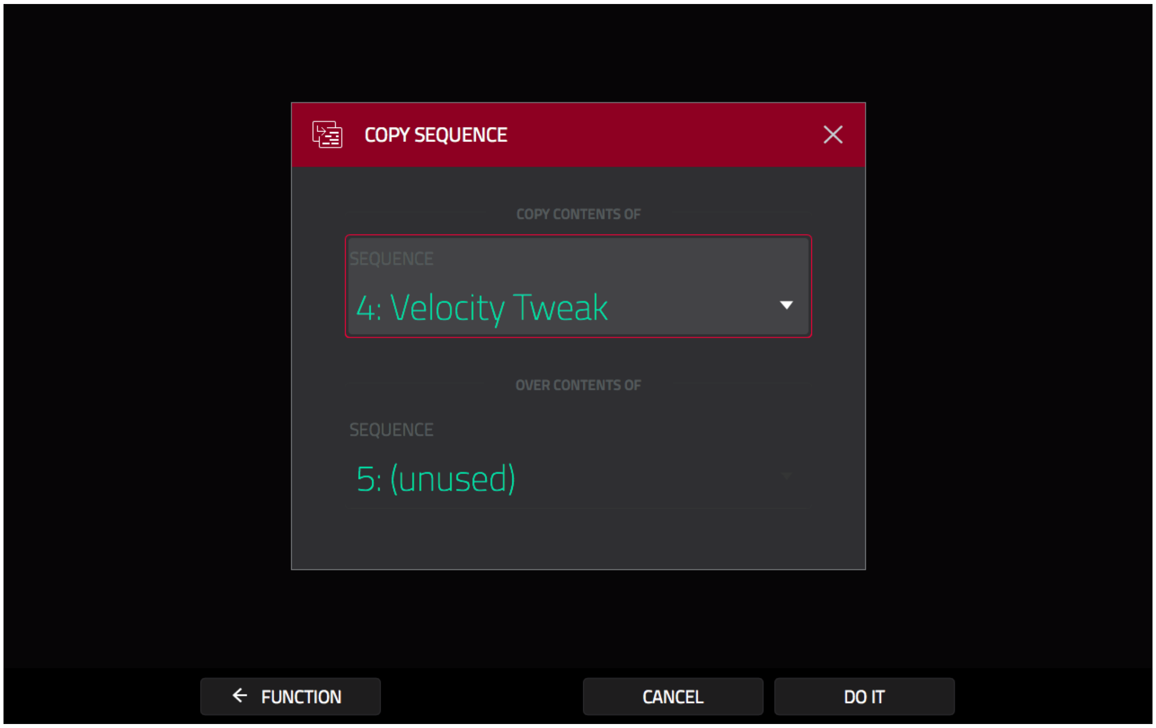
The beauty of timing correct is that there are many different time divisions (quantise templates) available, plus you are free to record some of your performance with timing correct on (for example, the main kicks, snare and hats), and then *overdub* additional notes with TC turned 'off' to add the more funky sounding notes afterwards, such as ghost snares and kicks.

Timing correct can be used at all points in your sequence creation - during a real time recording, during overdubbing, when manually adding events in the grid or you can even apply timing correct to an entire track.

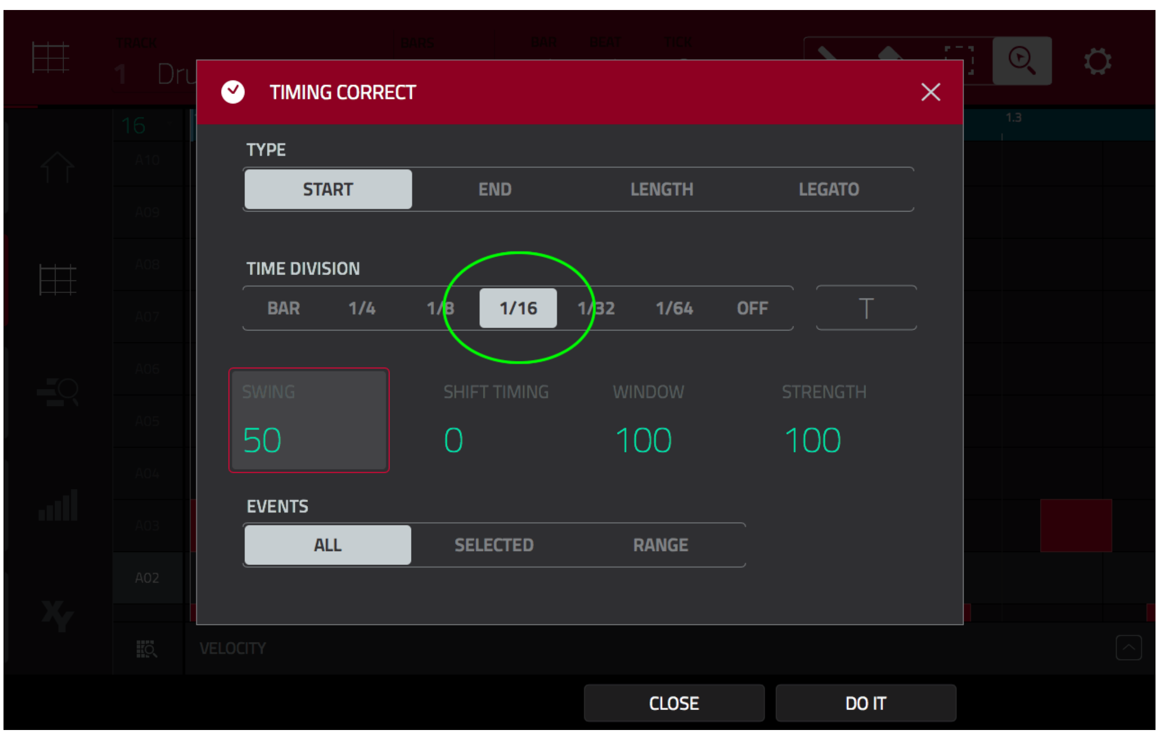
First make an exact copy of the currently selected sequence (if you've been following this tutorial, it should be the '**Velocity Tweak'** sequence). Go to **MAIN** and touch the sequence edit tool (the pencil) at the end of the **SEQUENCE** line. Here you can select '**Copy Sequence**'.



In the resulting pop up, leave the default settings and hit **DO IT** – this will copy sequence 4 to the blank (unused) sequence 5.



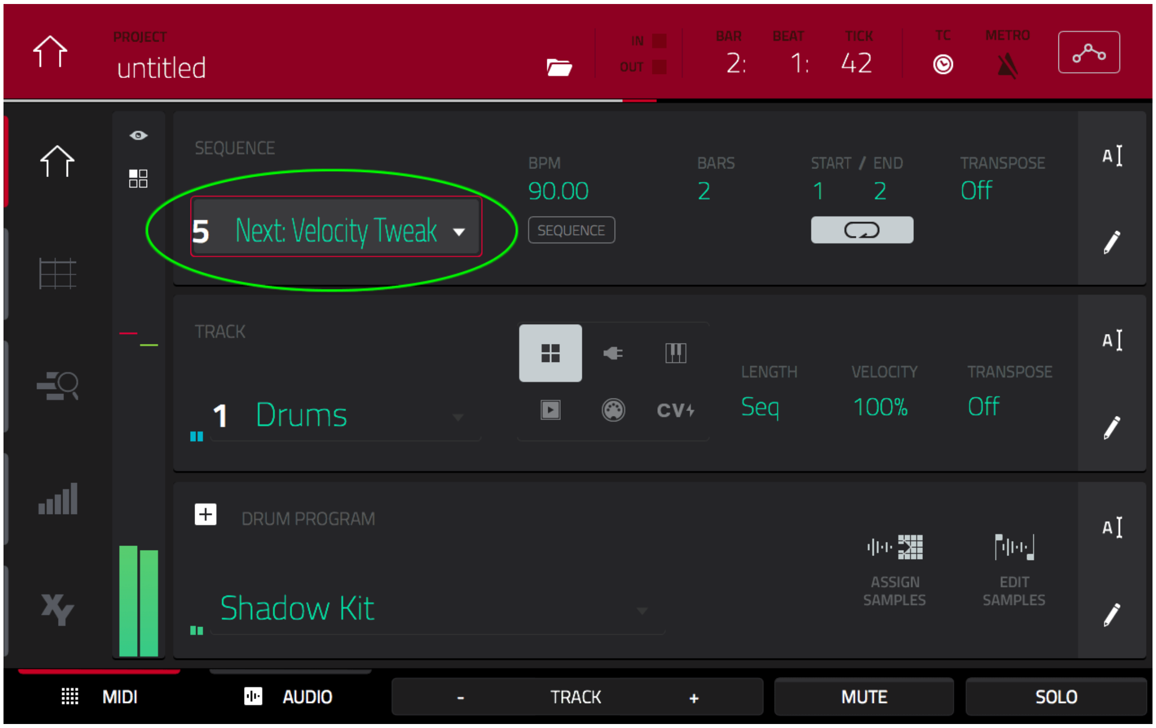
Rename the sequence '**Timing Correct 1-16**'. In your MPC, go to **GRID EDIT**, hold down **SHIFT** and press **T.C.** to bring up the **Timing Correct** screen.



Make sure '**TYPE** is set to '**START**'. This tells the MPC that we want it to align the start point of each event to the nearest quantise point. Now set '**TIME**

DIVISION' to **1/16** and '**EVENTS'** to '**ALL**'. Hit **DO IT** and the MPC will apply a quantise template to your track where all events are moved into a 1/16 time division grid.

Go to **MAIN** and press **PLAY START** to listen to the new 'tighter' version. While it is playing, touch the sequence name to select it and turn the data wheel one click anti clockwise:



Notice how the sequence name changes to '**Next: Velocity Tweak**'? This is the basic 'next sequence' functionality – it allows you to 'line up' the next sequence to play while the current sequence is playing back, ensuring the switch occurs without any pause and perfectly in time. Once sequence 4 (Velocity Tweak') starts playing you can 'line up' sequence 5 to play by turning the data wheel one click *clockwise* (or use the '**+**' button).

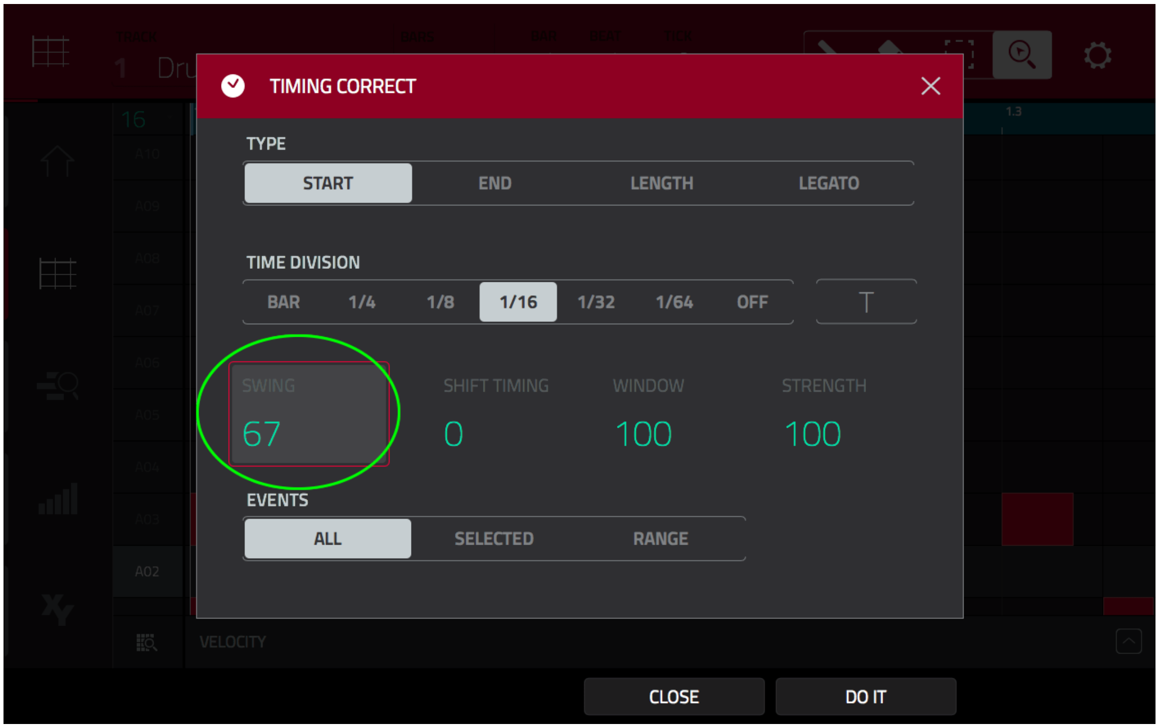
Listen and compare the 'before' and 'after' versions of the drum beat. The new sequence is tighter and more rigid in feel, while the original version is more loose and natural sounding. I wouldn't say either is better – it just depends on the feel you are wanting to achieve in your beat.

You can also apply other timing correct templates to your beat. Go back to the original '**Velocity Tweak**' sequence and copy it again. This time apply a **1/8 time division** – load up my version **Timing Correct 1-8.sxq**. The sequence is now even more rigid and has really stripped down the beat and removed most of the funkiness.

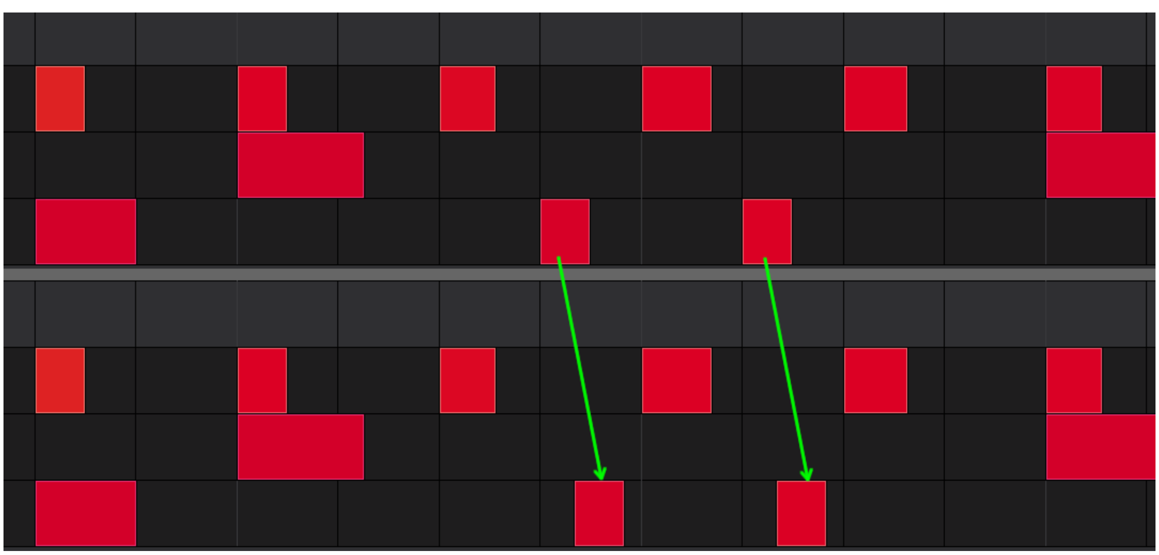
Applying Swing

Obtaining a vibe somewhere between the more rigid sound of a quantised beat and the looser feel of a beat recorded with Timing Correct 'off' is possible by using the **swing** parameter. Swing will add a nice but subtle groovy element to your beat, quite literally giving it more swing and a bit more bounce.

Once again, return to the original '**Velocity Tweak**' sequence and make another copy. Rename this copy **Swing 67**. In **TC** set a **TIME DIVISION** to **1/16** but this time set a **SWING** value of **67**.



Hit **DO IT**. Compare this new sequence with the '**Timing Correct 1-16**' sequence (which was also 1/16 but had no additional swing added). While most of the events are identical, there are small differences, for example kick events have been moved forward a little:



These subtle changes definitely give the beat a more bouncy feel. Now of course all of this could be achieved by finger drumming your beat with timing correct 'OFF', you would just have to play the beat naturally with that swing in the first place. However the swing function simply gives everyone the ability to quickly create a bouncy beat without the need to be the world's greatest finger drummer.

It's important to realise that timing correct and swing can also be applied *during* the recording and overdubbing process, and not just after recording. Your most recent timing correct settings are remembered in the TC screen, so if you were to record a sequence now, the MPC will apply both the timing correct and swing 'template' to your live recording. So if you were to hit pad A01 at a sequence time of 001.01.**007**, the MPC would record that event at the time of the nearest quantise point of 001.01.**000**.

So timing correct is a really great tool, but use it wisely when live recording as your original 'loose' feel will not be captured at all if Timing Correct is set to anything other than OFF.

Adding Live Feel to a Quantised Beat

A common recording procedure when making electronic music is to lay down the bulk of a track using a quantise template and then make some careful changes to add more natural feel and groove.

Let's return the timing correct values back to their default settings. Set **TC**:**OFF'** and change **swing** back to **50** (neutral swing).

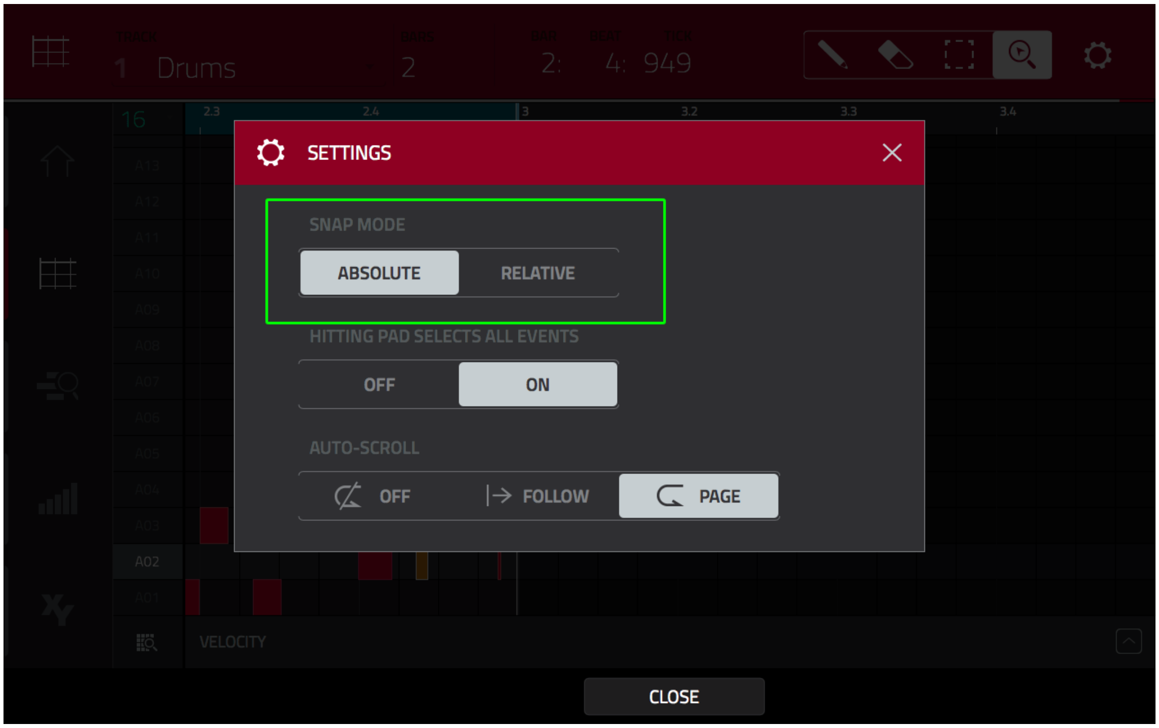
With Timing Correct '**OFF'** you can now overdub additional events to your sequence and they'll be recorded exactly as you played them. For example, try adding some ghost 'snares', which are subtle snare hits that give your beat a bit more groove – to record your ghost snares, hit **OVER DUB** and **PLAY START**, and remember to turn off **FULL LEVEL**. If you make a mistake, either hit UNDO or you can just manually move or delete the rough events individually afterwards.

Load up the sequence **Combo.sxq** for my version containing both sequenced and live events.

Moving Events When T.C. is Not 'OFF'

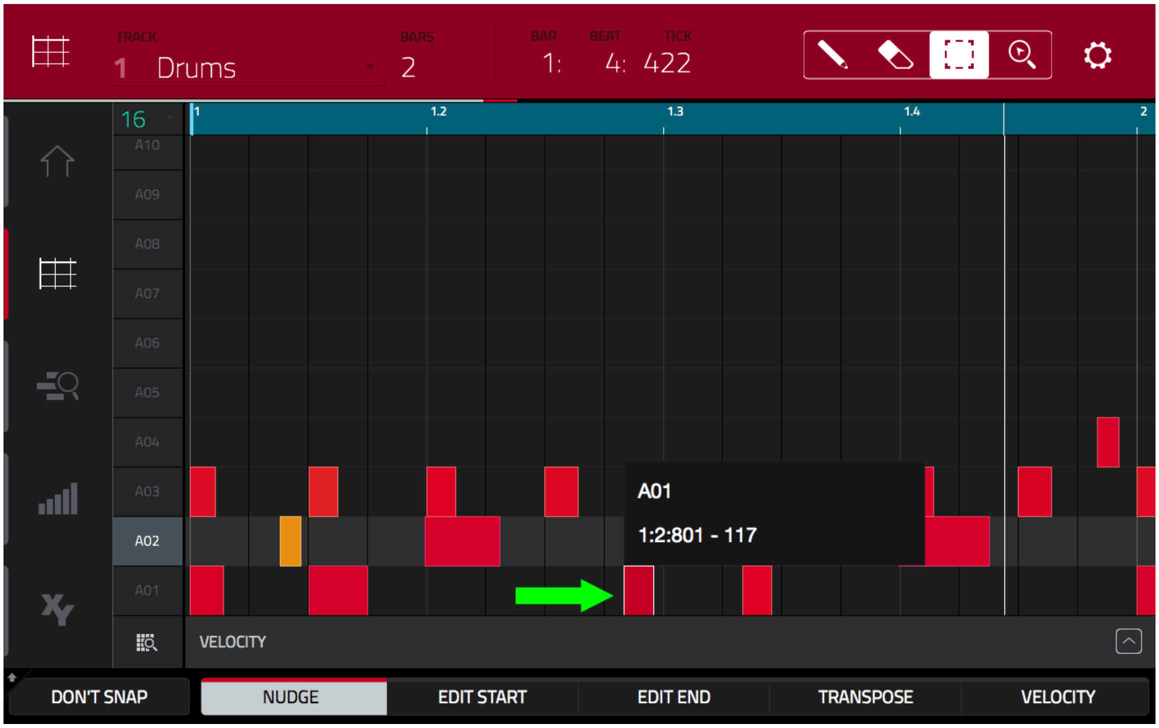
We've previously seen that you can manually move events using the 'select' tool, but when we did this previously, **TC** was set to **OFF**. However, if TC is not 'OFF' then the MPC will (by default) limit the position to which we can manually move an event.

Select a **T.C**. of **1/16** and hit the '**grid settings**' gear icon at the top right of the screen:



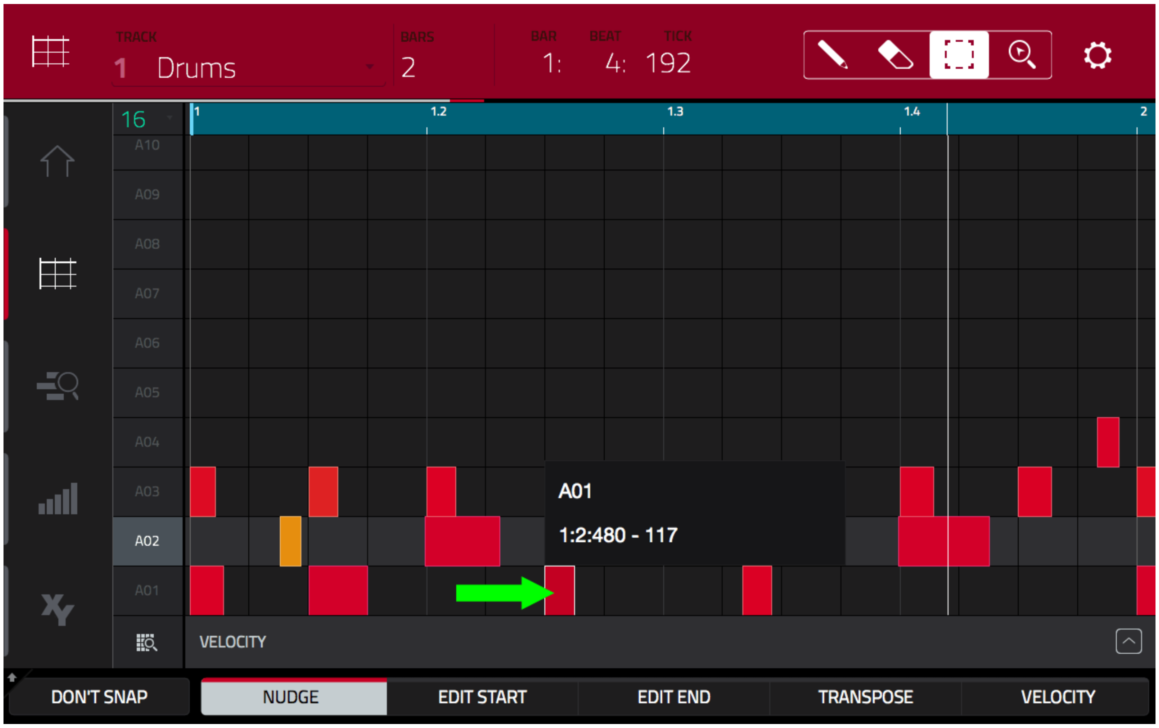
The top option allows you to set the **SNAP MODE**. The default setting is '**ABSOLUTE**' and it's the one we've used so far in this book. In 'absolute' mode, moving an event will 'snap' the event to the nearest position in the current T.C quantise template.

Using the 'select' tool, select the kick at time **1:2:801**.



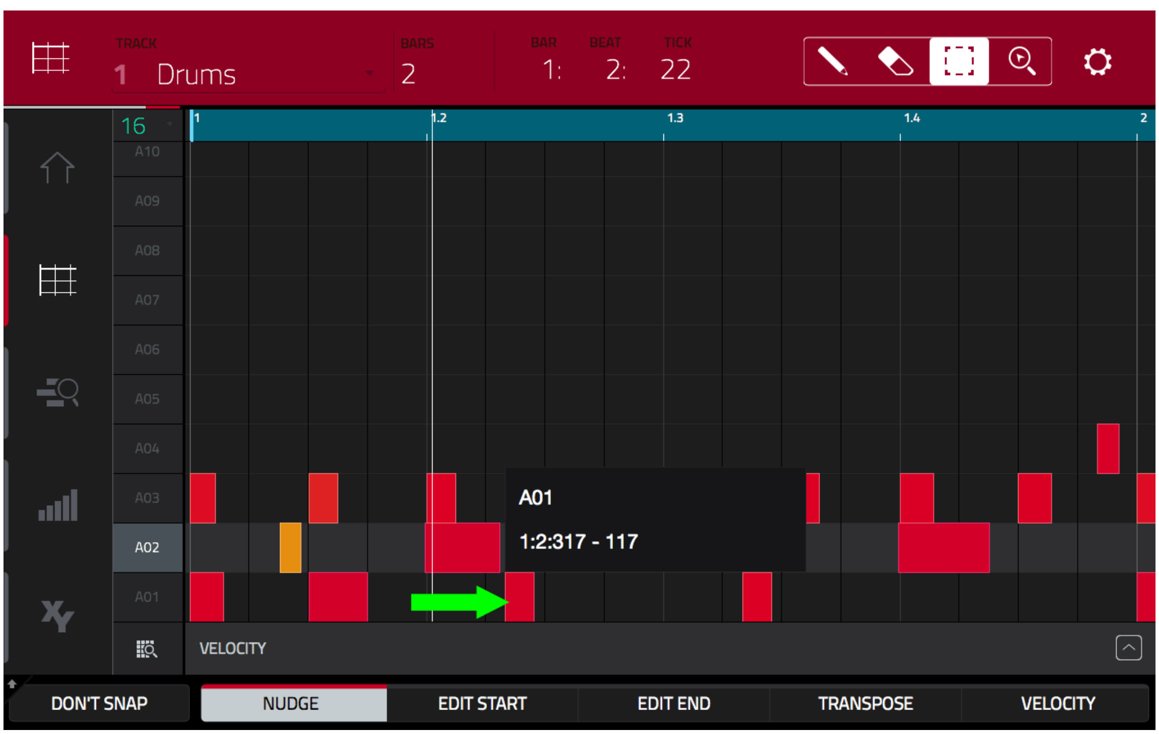
This kick is currently at 801 'ticks' into the second beat of the first bar which is outside of the 1/16 quantise grid (1/16 quantise points are at 000, 240, 480 and 720). It's placement here helps give the beat that almost lazy, swing feel.

With a **SNAP MODE** set to the default **ABSOLUTE**, drag this kick two steps to the left:



With ‘absolute’ snap, the event has moved to a 1/16 quantise point, **1:2:480**. Hit **PLAY START** to hear how it sounds.

Hit **UNDO** to take the event back to **1:2:801**. Now go back to **GRID settings** and change **SNAP MODE** to **RELATIVE**. Now drag the event to the left:



This time the event does not move to the normal 1/16 quantise points of 720 then 480, instead it moves to **797**, **557** and **317**.

What's happened? With 'relative' snapping, the event still moves in a 1/16 grid, but that entire grid is 'offset' to account for the distance the original event was from the nearest quantise point. Press **PLAY START** to hear the 'relative' snap and you should hear that it's retained that 'lazy' feel.

Relative snap is good for moving around events that were originally 'unquantised', this way by moving them you will not loose the original groove of those events as they remain 'relative' to the off-grid groove they were originally played with.

It's likely that for the majority of grid work you'll probably find that 'ABSOLUTE' is the most used setting. But another option is to temporarily disable the quantise grid and go completely 'freestyle'. To do this, hold down the **DON'T SNAP** button at the bottom left of the screen while you drag (or NUDGE) an event around.

The 4 tick Limitation in GRID EDIT

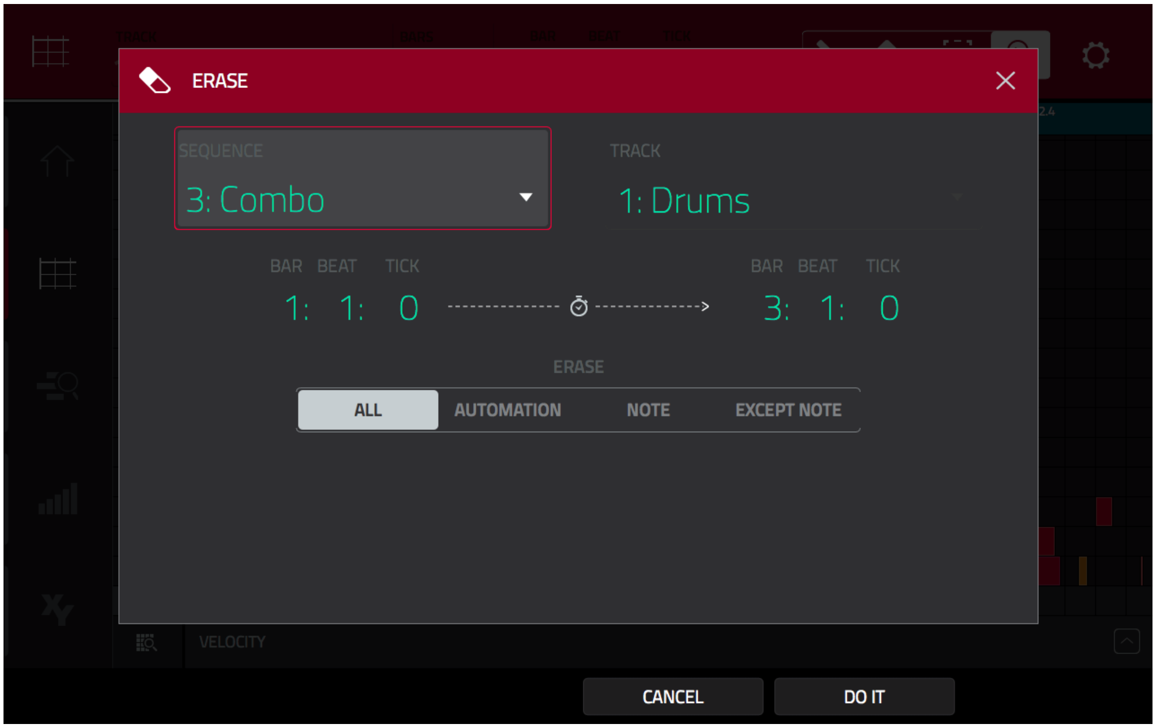
Please note the smallest movement increment available in GRID EDIT is 4 ticks. If you do the maths for our 'relative' snapping example you'll spot that the final location is actually 4 ticks 'out' from what the true 'relative' position should be (it should be 321, not 317) - I assume this is related to the 4 tick limitation. It's unclear why Akai do not allow single tick moves.

Erasing Events

If you don't like a particular event in the grid you can easily erase it and there's two ways to do this directly in the GRID; firstly if you have the **pencil** tool selected, just *quickly* **double tap** the event you wish to erase. Alternatively, select the **eraser** tool:

With the eraser tool selected, just **single tap** an event to erase it or drag the tool across a range of events to remove them all.

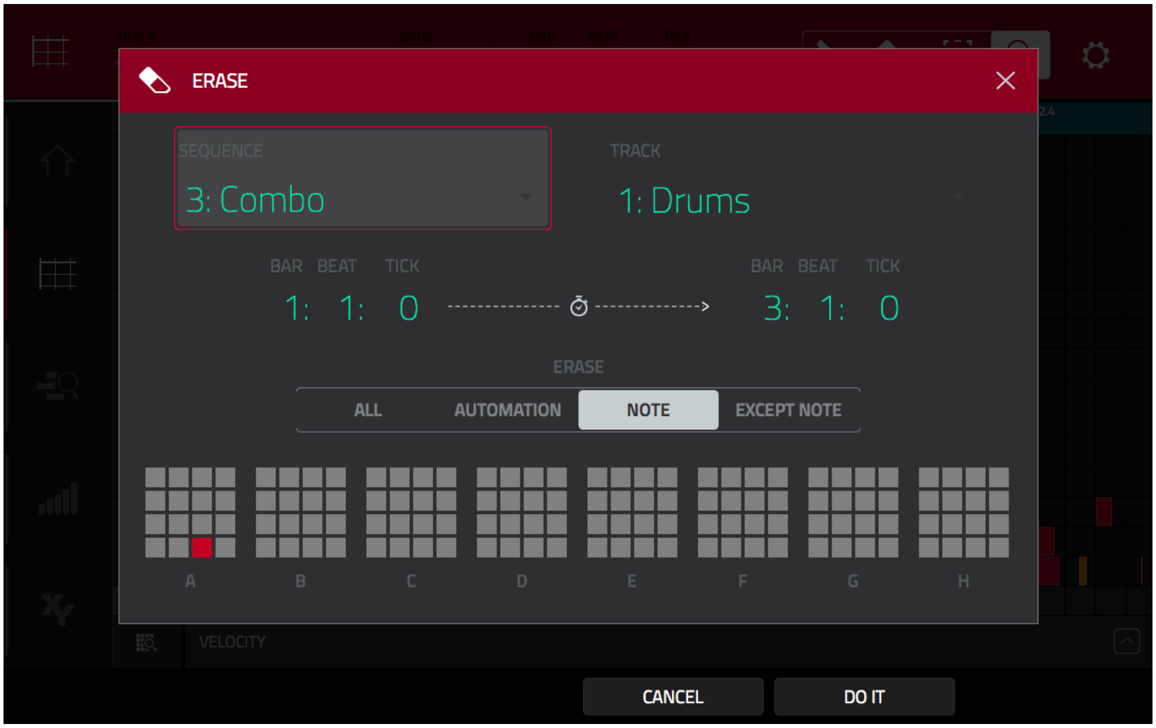
An alternative is to hit the dedicated **ERASE** button on your MPC hardware.



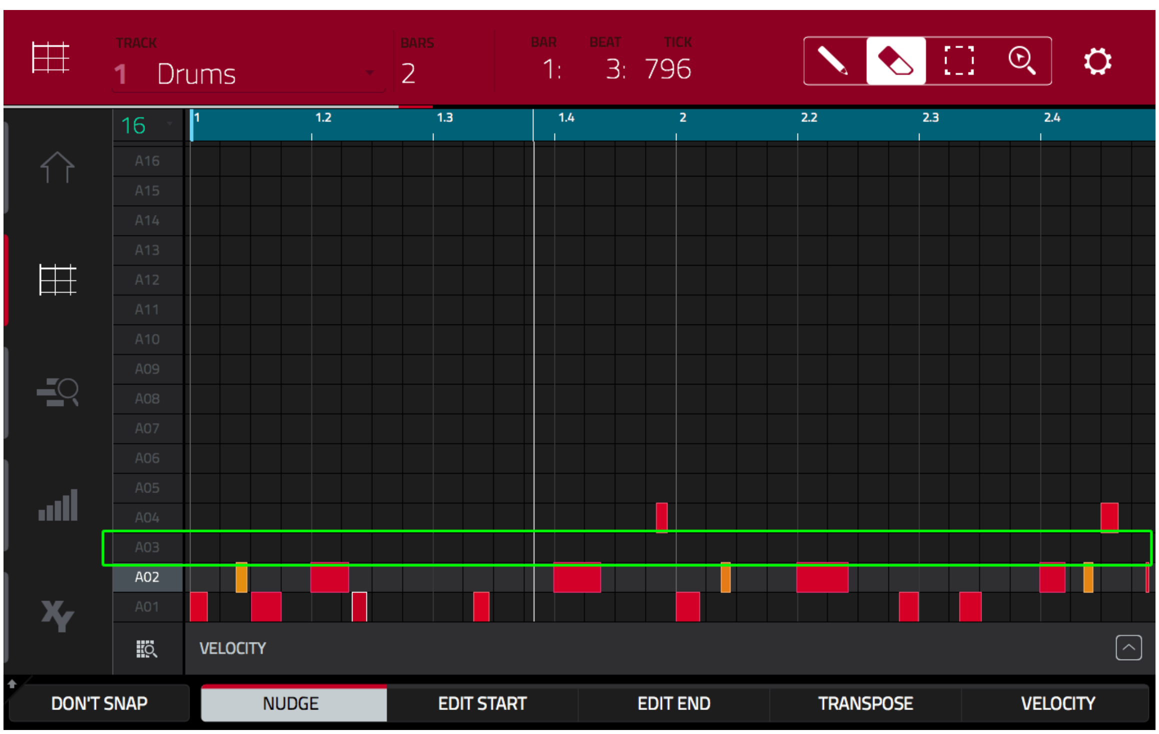
The **ERASE** screen lets you erase events on the currently selected track across a specific time range. By default it will cover the entire active range of the current track, but you can change this if you need to. Leave the **range** as it is (**1:1:0 to 3:1:0**).

The second option lets you chose which types of events will be erased. The default is '**ALL**' which will erase every single event on the track (across the chosen time range), which we definitely do not want to do as this would also erase our kicks and snares.

If you wish to only delete specific pads, select the **NOTE** option. Just hit the pads you wish to erase (hit a pad again to remove it from the list) – so hit **A03**:



Hit **DO IT** and you'll see all your A03 hi hat events have been removed from the sequence:



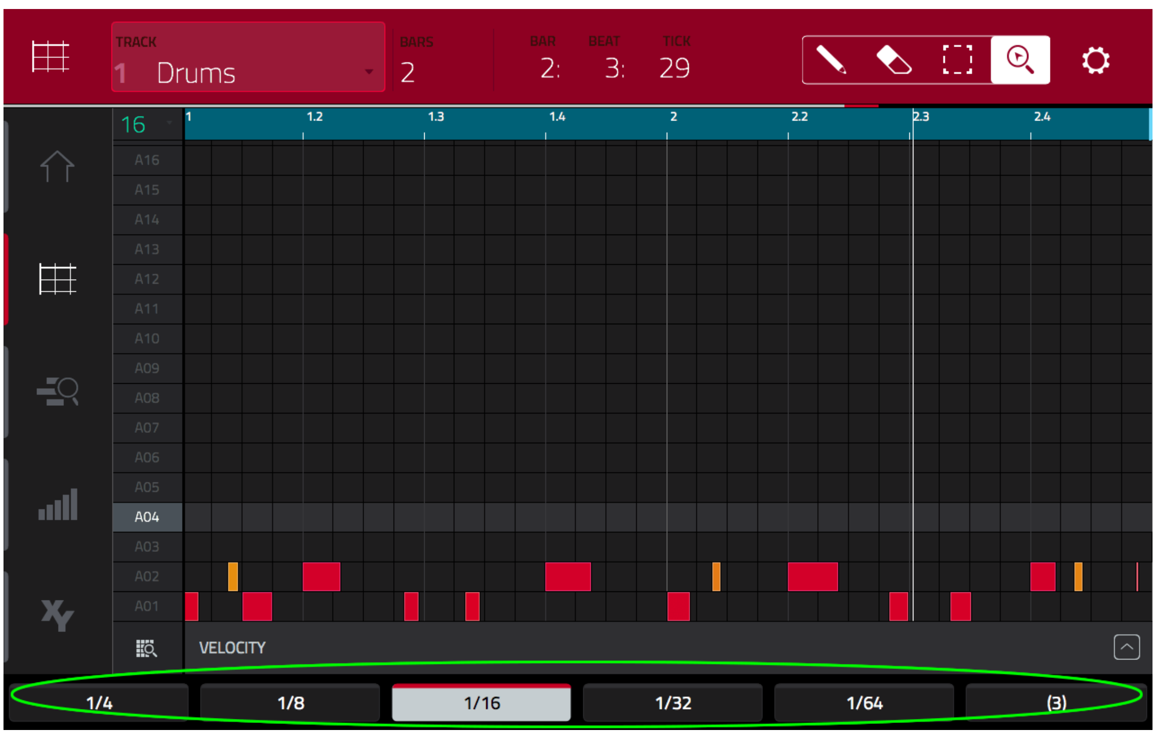
ERASE in Real Time

You can also erase notes on-the-fly while you are recording. Hit **PLAY START** and **OVERDUB**, hold down the **ERASE** button and then press and hold pad **A04** (our open hat). When the play head passes an open hat event it will erases the open hat from your sequence (you can also hold down more than one pad if you wish to erase multiple pads).

Recording Hi Hats With Note Repeat

NOTE REPEAT is a legacy MPC feature that allows you to easily record multiple quantised events in real time, and is perfect for recording hi hat patterns.

Press **PLAY START** in our now 'hat-less' sequence and leave it playing in a loop. While it is playing press and hold down the **NOTE REPEAT** button and you'll see a new row of buttons appear at the bottom of your GRID:



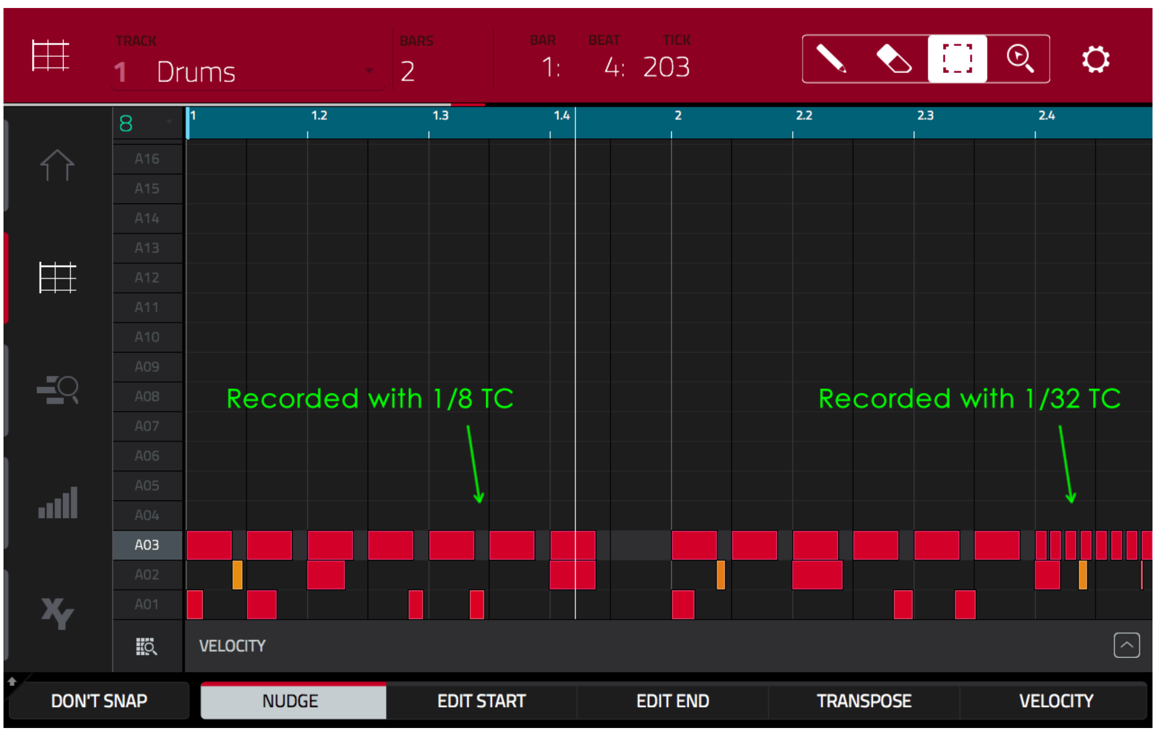
Here you can quickly change the quantise '**TC**' setting. Set it to **1/4**. Continue to hold down **NOTE REPEAT** and now hold down pad **A03**. As you can hear, NOTE REPEAT repeats the sample at regular ¼ bar intervals. Change the TC setting to **1/**8 and again hold down **NOTE REPEAT** and **pad A03**. Now the hat is repeated at 1/8th bar intervals (as it was originally before we deleted it).

Experiment with the different TC settings – you can also press the **(3)** button to select triplet feel (for example with '3' and 1/16 selected, you'll get a TC of 1/16(3)). *Please note that NOTE REPEAT does not work with TC Set to OFF.*

If you turn off **FULL LEVEL** and vary how hard you push down on pad A03 you'll hear the velocity change accordingly so everything remains fully velocity sensitive in NOTE REPEAT mode.

When you are ready to record, choose any TC setting, press **OVERDUB** and your sequence (which should be already playing) will enter OVERDUB mode. Your 'note repeat' hats will now be recorded to your sequence. Remember you don't have to hold down the NOTE REPEAT button constantly, you can release it to create additional gaps in your hi hat pattern.

You can also of course 'mix and match' by overdubbing your hats using different TC settings within the same track. Load up the sequence **Note Repeat.sqx** and hit **PLAY START**.



I recorded most of the sequence using a **1/8 TC** - note the brief pause at the end of the first bar – to create this I just took my finger off pad A03 temporarily. At the end of the second bar I used a **1/32 TC** to produce a rapid hi hat roll.

Latching NOTE REPEAT

If you hold down **SHIFT** and hit **NOTE REPEAT** you put NOTE REPEAT into ‘**LATCH**’ mode. While latched, NOTE REPEAT does not need to be held down and any pad press will get repeated as per your TC settings.

Correcting Mistakes with UNDO

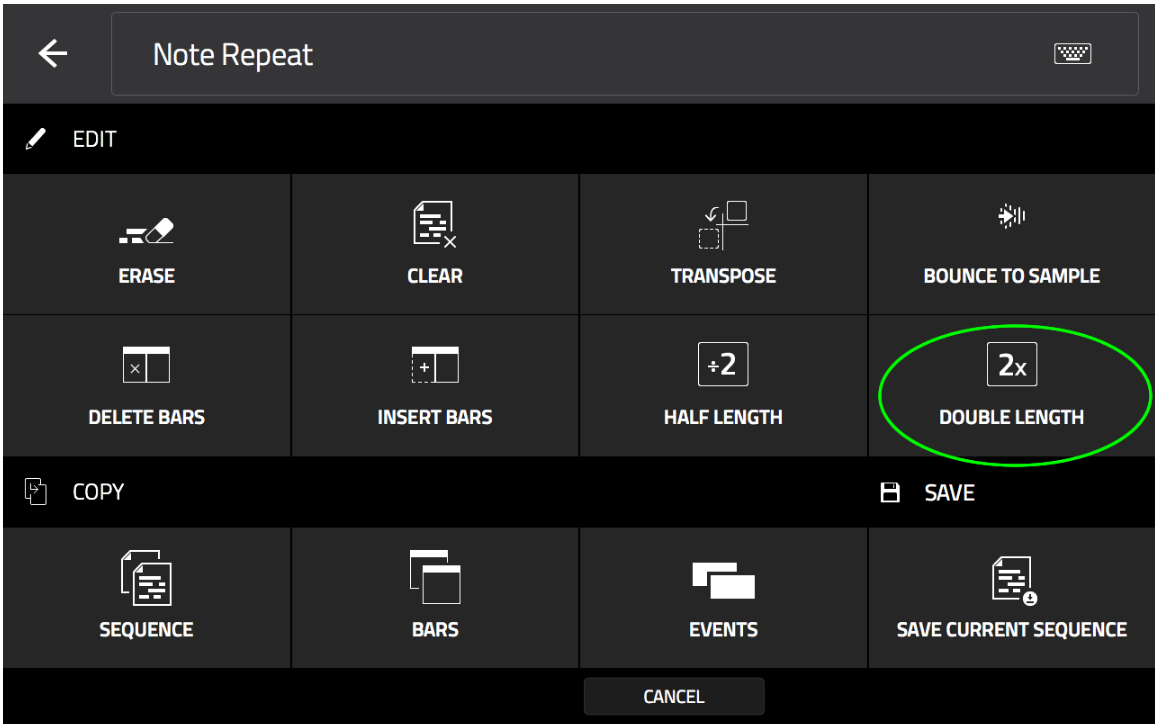
If you make a mistake, don’t worry, you can 'undo' most actions by simply hitting the dedicated **UNDO** button on your MPC hardware. So hit **UNDO** once to undo the last action, hit it 10 times to undo the last 10 actions, and so on. Hold down **SHIFT** and press **UNDO** again to ‘**redo’** any edits that you previously ‘undid’ (i.e. undo the undo!).

Methods For Copying Events

Currently our beat is 2 bars long which is okay for sketching out an initial idea but you might begin to find working within a small sequence a bit limiting and somewhat monotonous. Later in the book we'll look at how we can combine multiple sequences into a complete song, however we can easily extend our current sequence so we can add a bit of variety to our beat.

Double Length

In **MAIN**, hit the **top row pencil icon**:



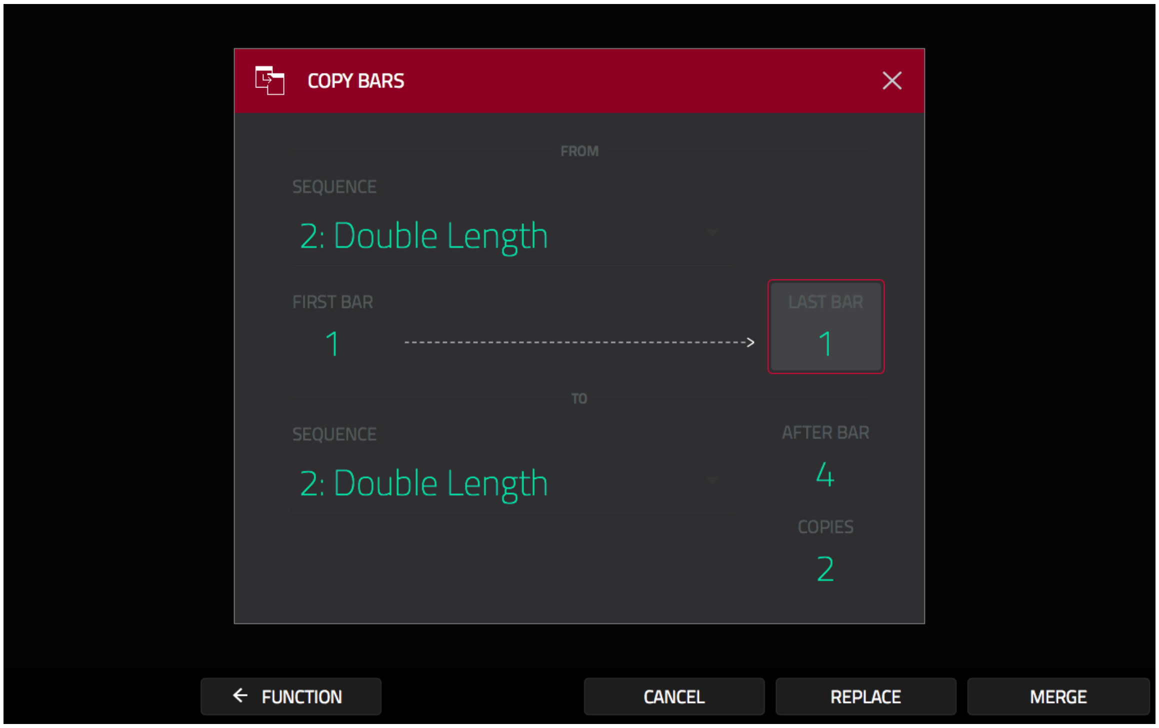
Hit **DOUBLE LENGTH** and return to **GRID EDIT**. At the top of the screen you should see that the sequence length is now listed as 4 bars - hit **PLAY START** and you'll see that the MPC has copied and pasted all the events from the original sequence to create a sequence double the length of the original. DOUBLE LENGTH duplicates the *entire* sequence; that's all events over all bars over all tracks.

With an additional 2 bars to our sequence we can use some of the techniques discussed in the previous chapter to make some small changes to our new bars 3 and 4. Load up the sequence **Double Length.sqx** which also contains some overdubs, erases, nudges and transposition to bars 3 and 4 using the techniques we've been covering in these sequencing chapters.

Copying Bars

You can also select *specific* bars to duplicate. Back in '**MAIN > SEQUENCE top row pencil icon**', hit **COPY > BARS.**

We can use this screen to copy a single bar or even a range of bars from any sequence. Let's copy just the **first bar** of our **Double Length** sequence. Configure the **COPY BARS** screen as follows:



Let's look at the settings. The **FROM** sequence is the sequence we are copying 'from', so this is set to our **Double Length** sequence.

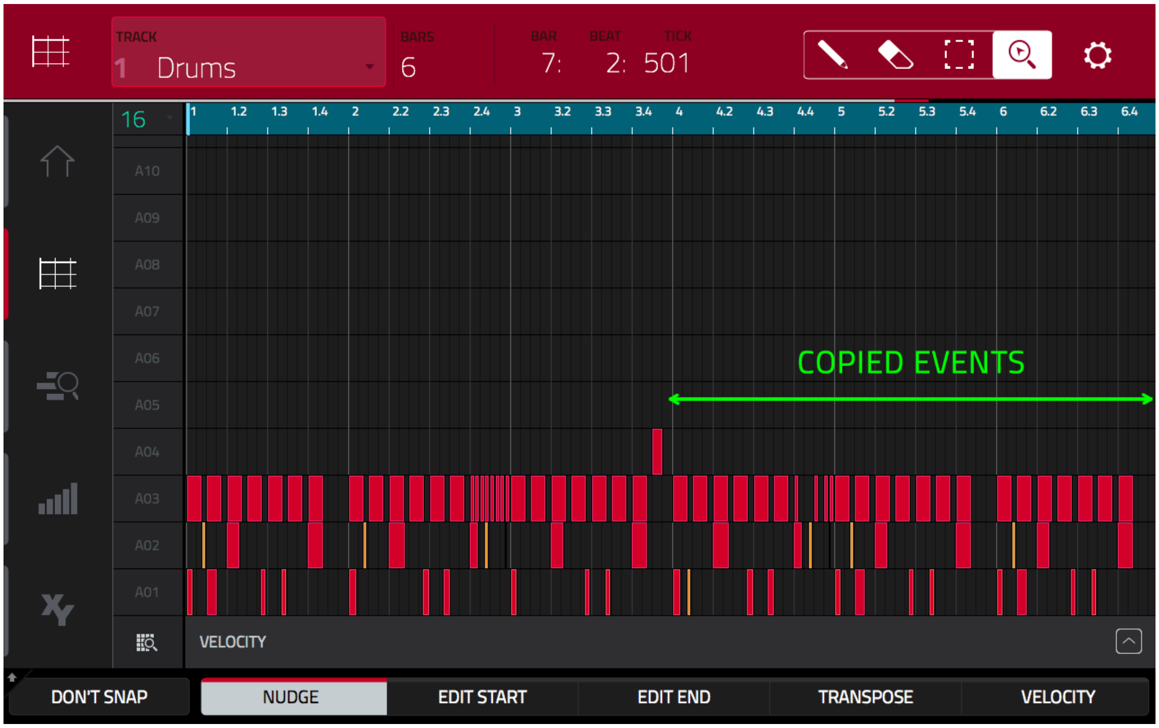
We now need to tell the MPC which bars to copy; we do this by setting a range of bars. In this case I only want to copy bar '1' so set **FIRST BAR** to **1** and also set **LAST BAR** to **1**. This will copy everything from the very start of bar 1 to the very end of bar 1 - all events, all tracks.

Now we have to choose the destination. This can be any sequence in our project, but I'm just going to place this copy of bar 1 at the *end* of our current sequence. Set **TO** to our current **Double Length** sequence.

AFTER BAR tells the MPC which *time location* you want the copied bars to be pasted. In this case I'm telling the MPC to paste after **bar 4** - i.e. at the end of bar 4. As our sequence is only 4 bars long, this will *extend* the length of our sequence.

Finally I've set **COPIES** to **2**. This will paste 2 copies of bar '1' after bar 4, extending our sequence to 6 bars long.

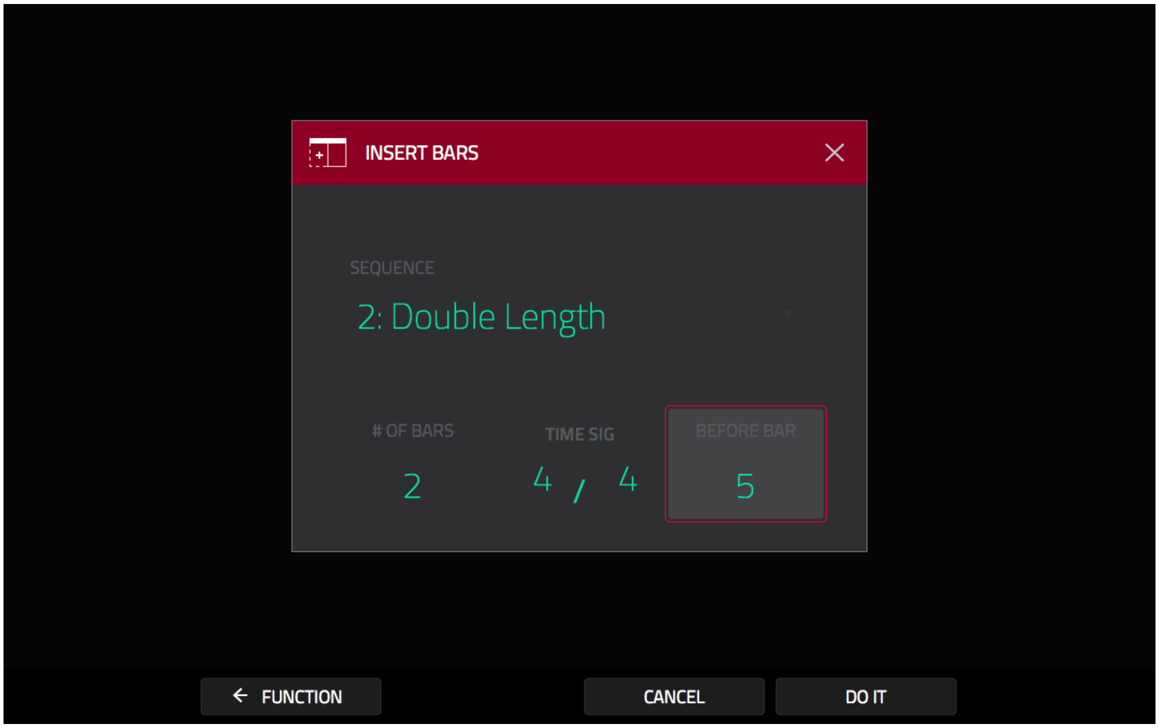
To copy the bars hit either **REPLACE** or **MERGE**. Return to **GRID EDIT** and zoom out to see the 6 bar sequence. Hit **PLAY START** to preview.



Inserting Blank Bars

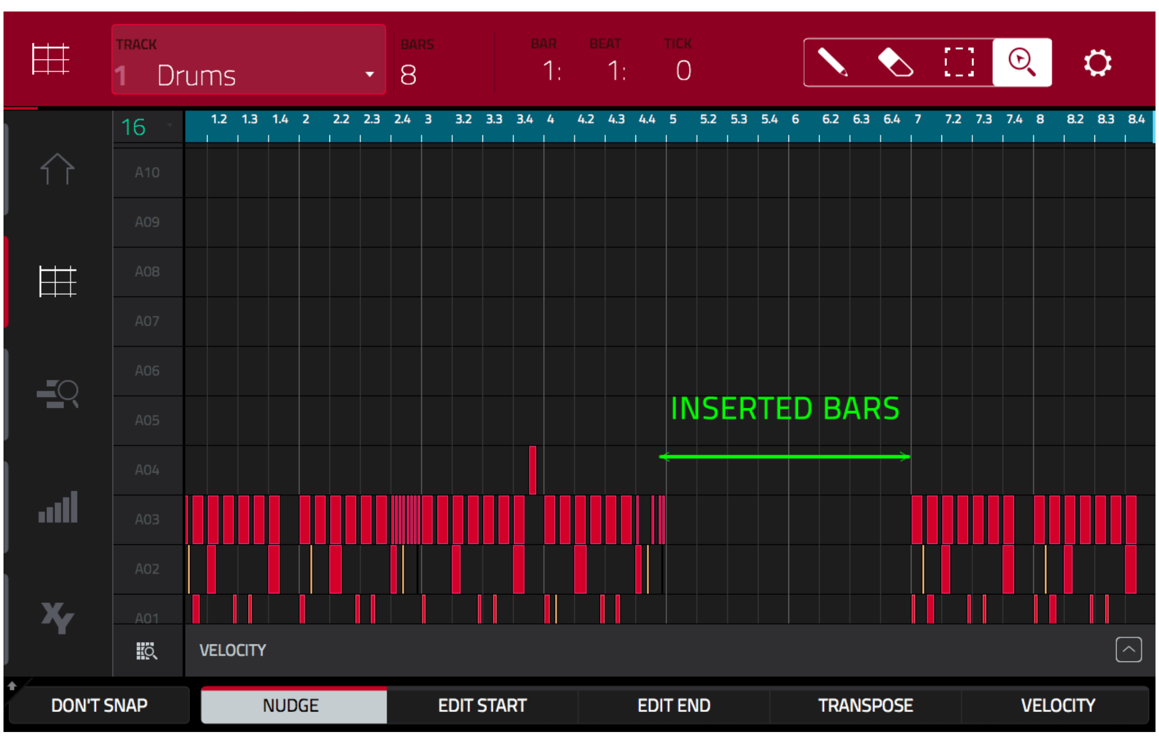
We can also insert blank bars into our sequence which will give us the opportunity to record completely fresh content inside an existing sequence.

Go to **MAIN** > **SEQUENCE row pencil icon > EDIT > INSERT BARS:**



With the settings above we will be inserting 2 blank bars (**# of BARS**) into the current sequence (**Double Length**). These two blank bars will be placed before bar **5** (**BEFORE BAR**).

Hit **DO IT** and take a look at our sequence in **GRID EDIT**:



As you can see our sequence now features a blank 2 bar segment in bars 5 and 6 - the sequence is now 8 bars long. Load up my version; **Insert Bars.sqx**.

Extending Sequence Length

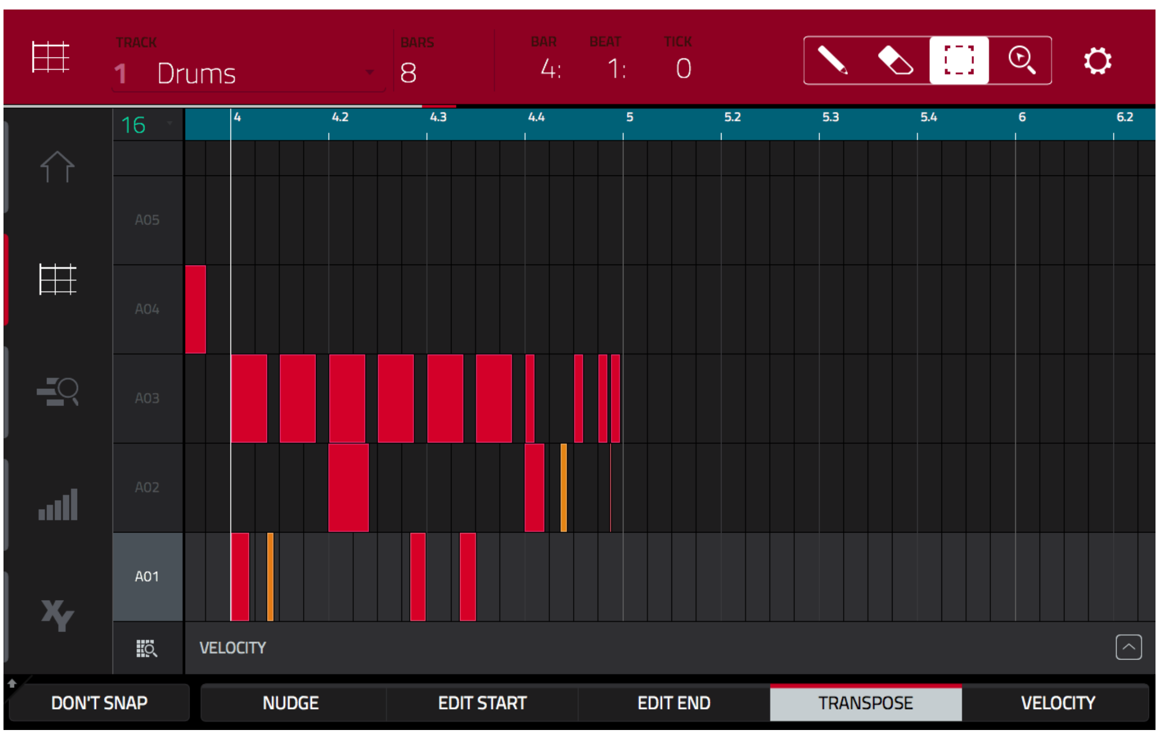
If you just want to add some blank bars to the end of a sequence, we can simply extend the entire sequence length by increasing the **BARS** parameter in **MAIN** or **GRID EDIT**.

Copying Events In GRID EDIT

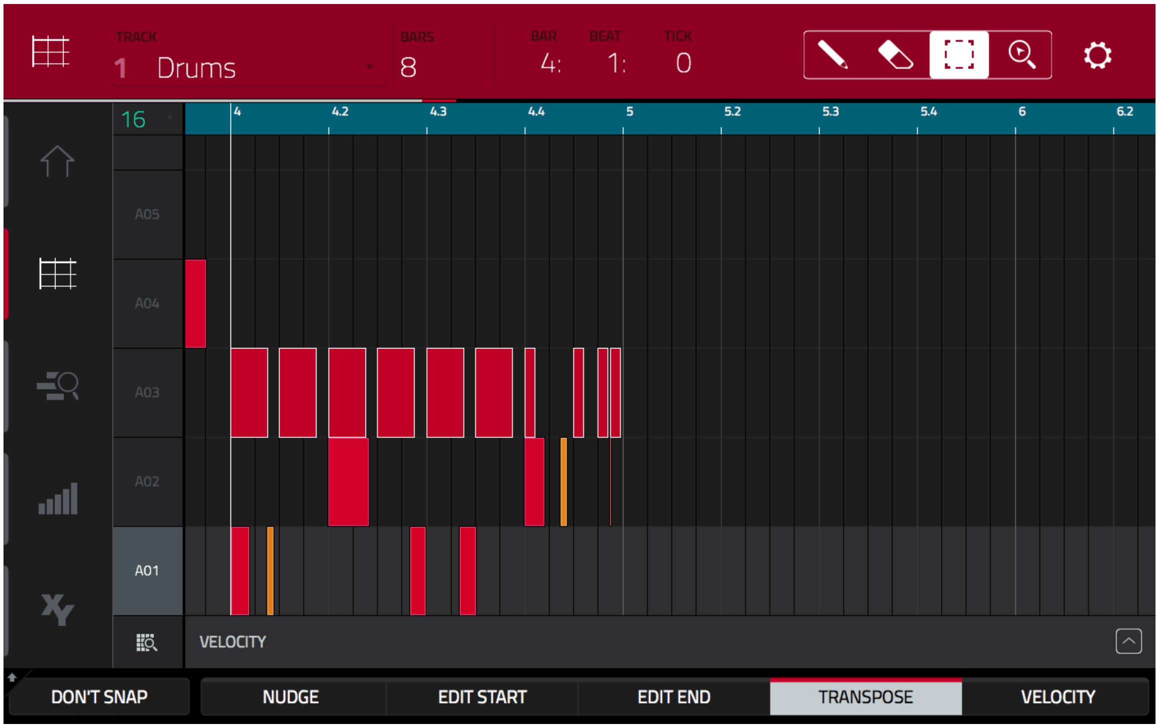
We can copy and paste a specific range of events directly in **GRID EDIT** and there's a couple of ways to do this.

Copy/Paste with the COPY button

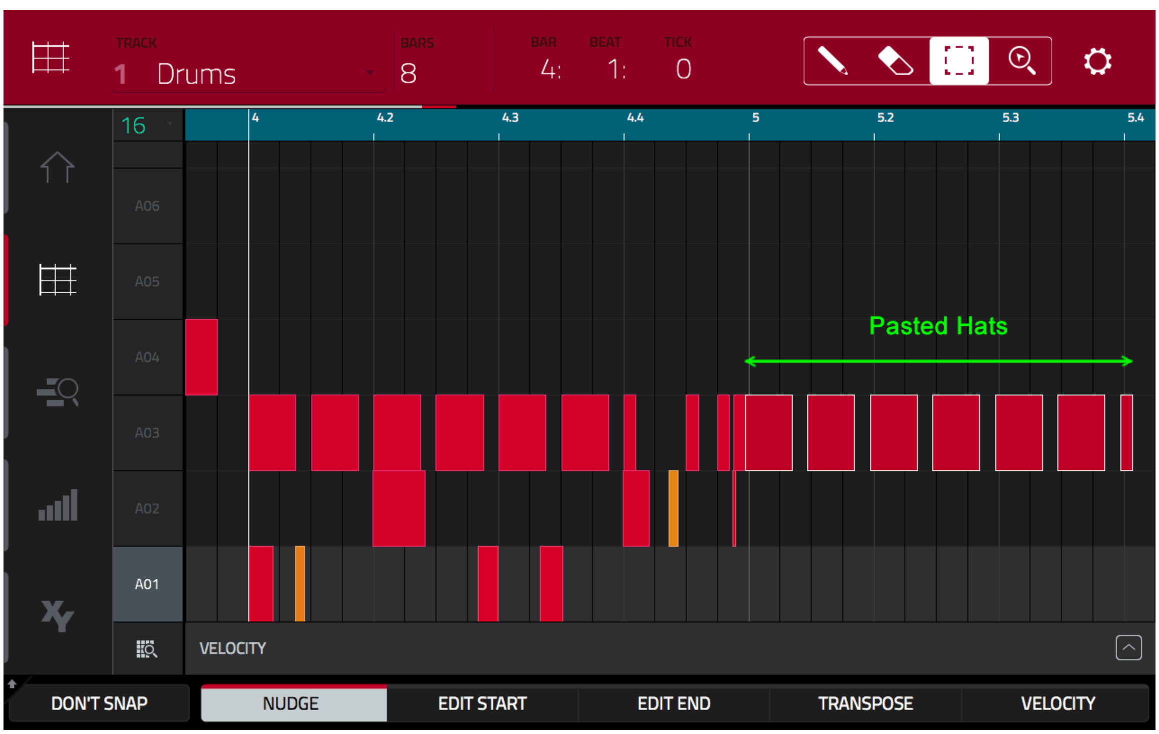
Let's copy the **hi hat pattern** from **bar 4** and paste it at the **start of bar 5.** First select the **zoom tool** and magnify the grid around the start of bar 4:



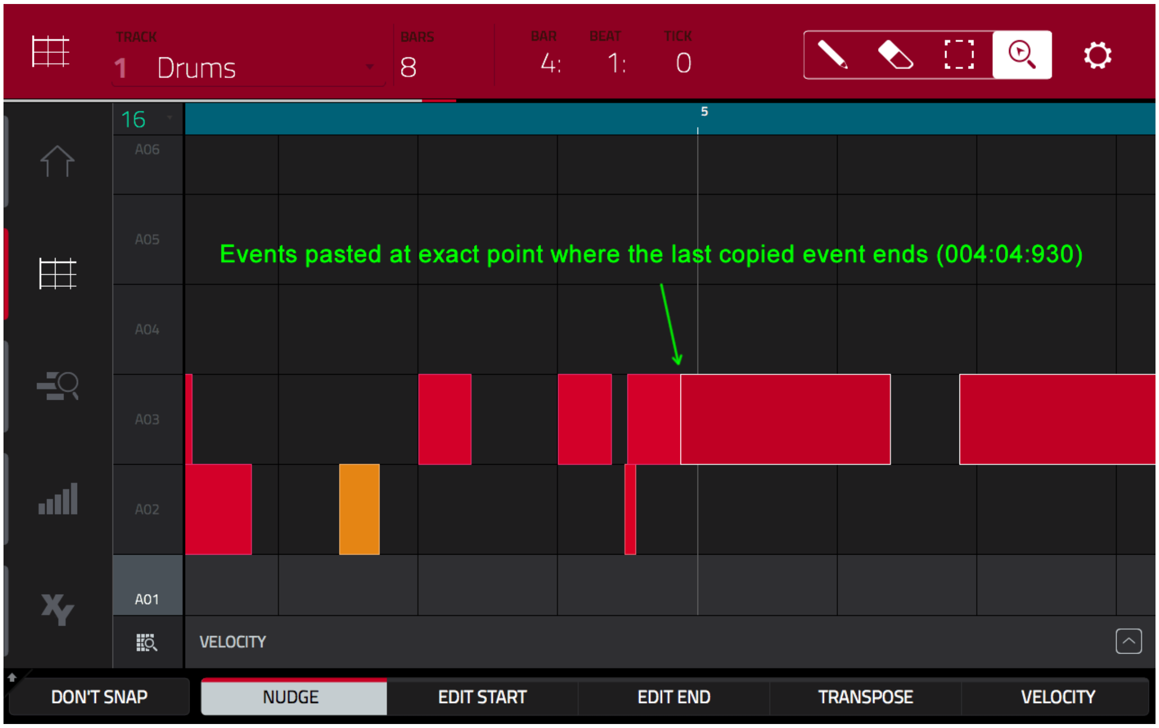
Now we must select the events you wish to copy, so choose the **select tool** and drag around the **closed hats** (**pad A03 row**) for the duration of **bar 4**:



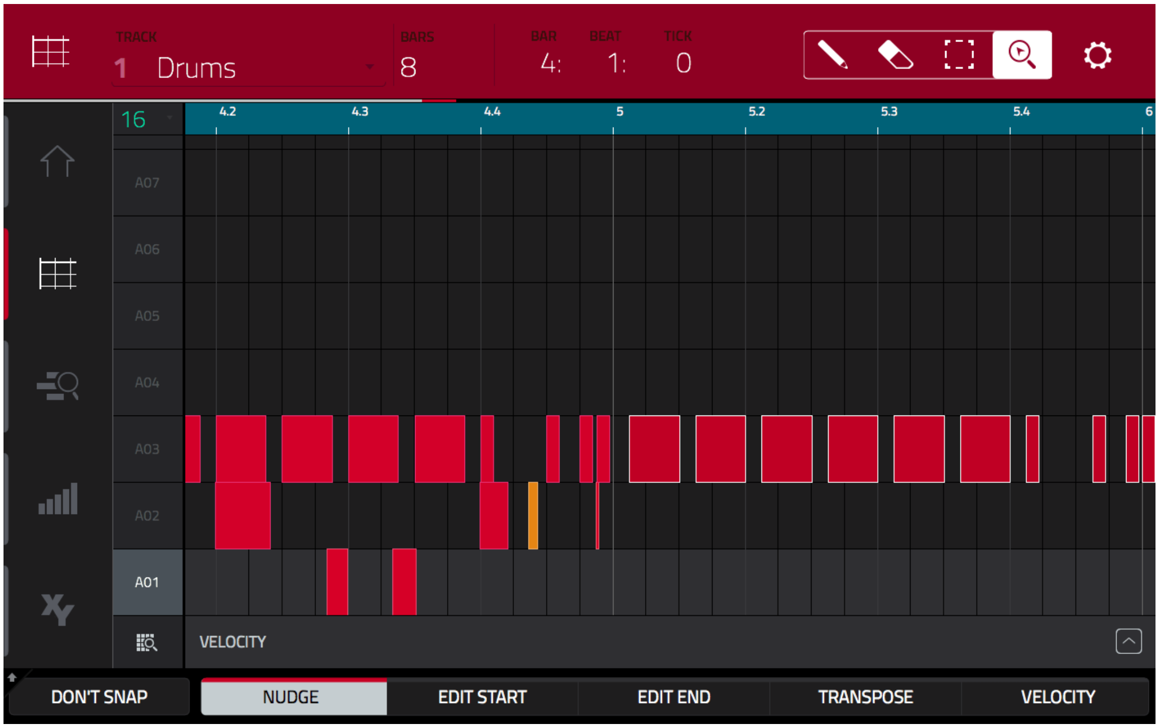
Now simply hit the hardware **COPY** button.



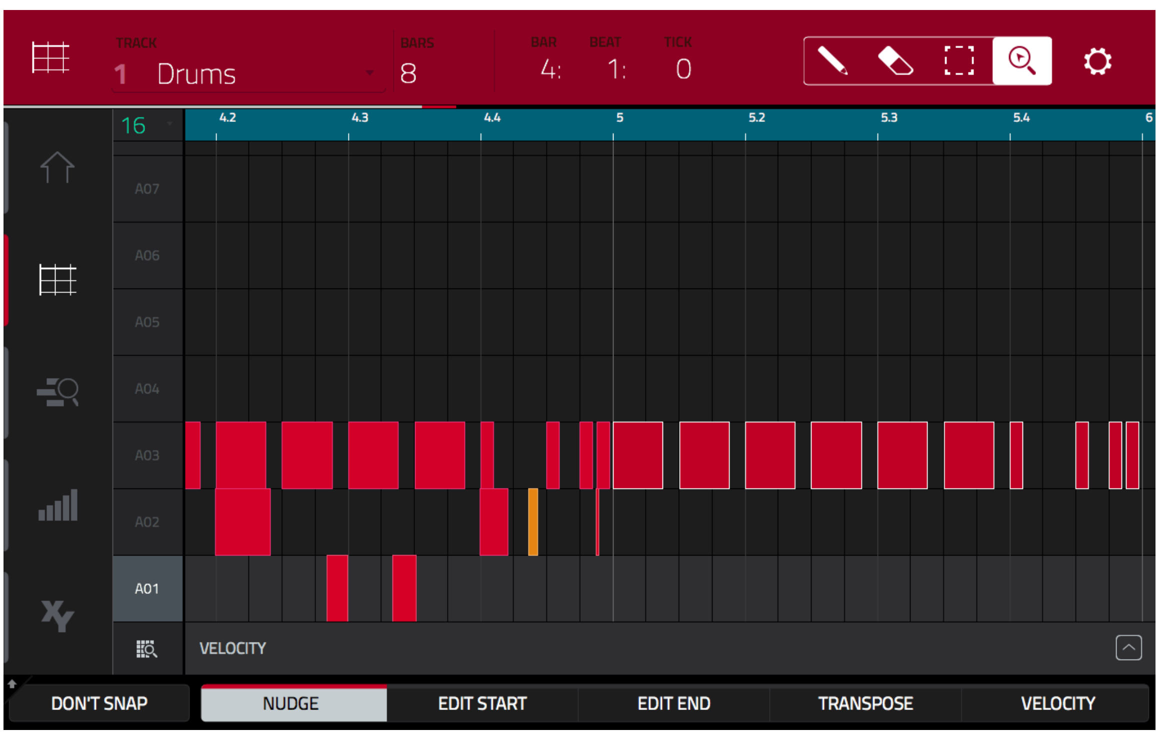
At this point the MPC copies the selection and simultaneously pastes the copied events literally at the end of the original selection. If you look closely you can see that the pasted events have actually been placed at the point where last event's duration actually ended, **004:04:930**.



So let's 'nudge' the pasted events so they start at exactly **005:01:000**. Press **NUDGE** and turn the data wheel **one click clockwise**:

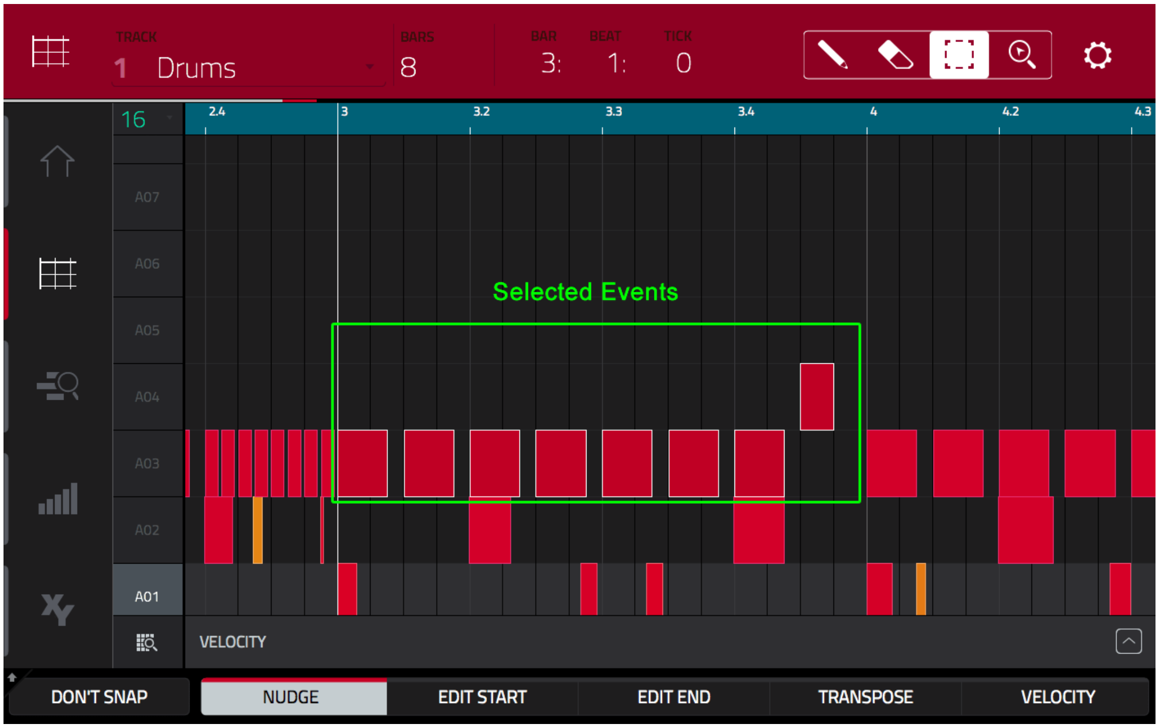


This has moved all the events too far, so now turn the data wheel back **one click anti-clockwise** and the events will 'snap' to the nearest 1/16 quantise point at 005:01:000 (you could also just drag the events with your finger).

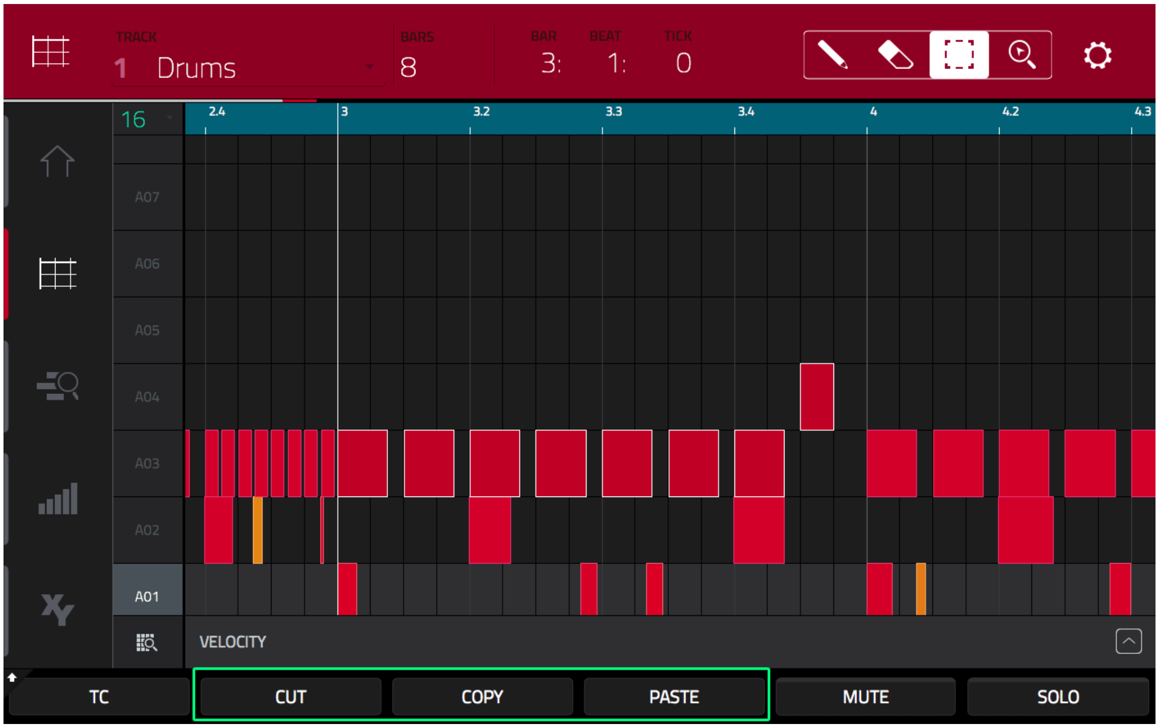


Copy/Paste Via SHIFT Buttons

Let's copy and paste the hats from bar 3 to the empty bar 6. Select the **zoom tool** and drag on screen so the bar 3 hats are in view. As before, choose the **select tool** and drag around to select the **closed hats (A03)** and **single open hat (A04)** of **bar 3**:

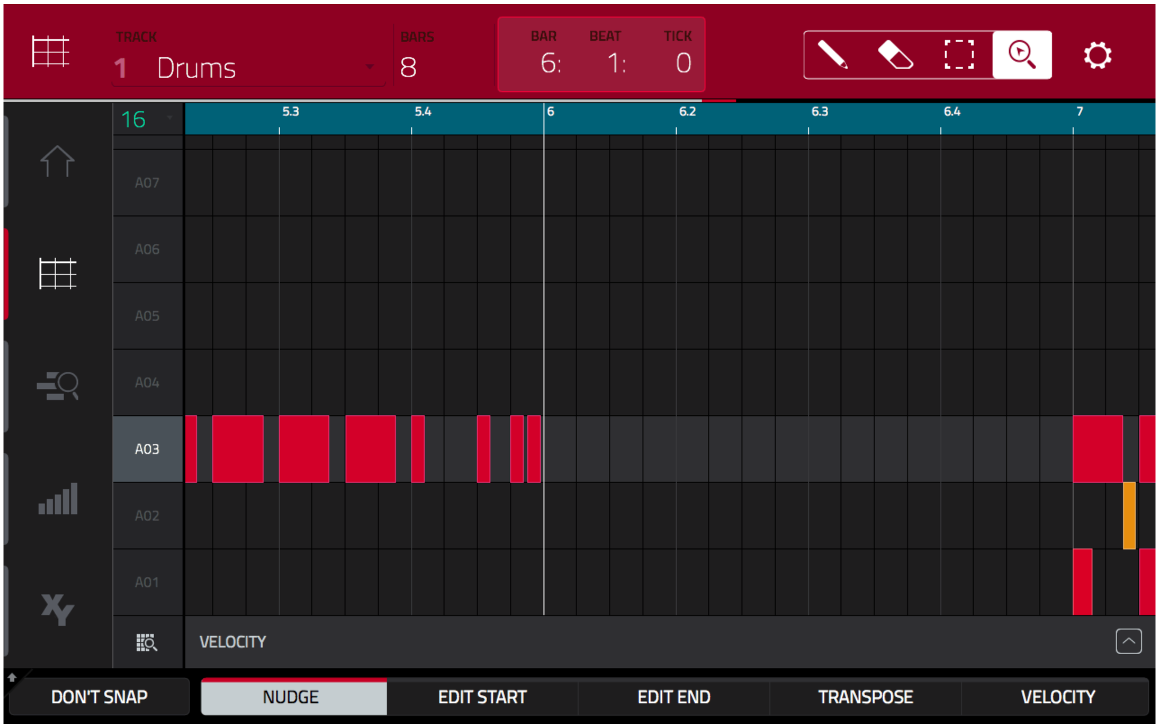


Hold down **SHIFT** to reveal the secondary menu:

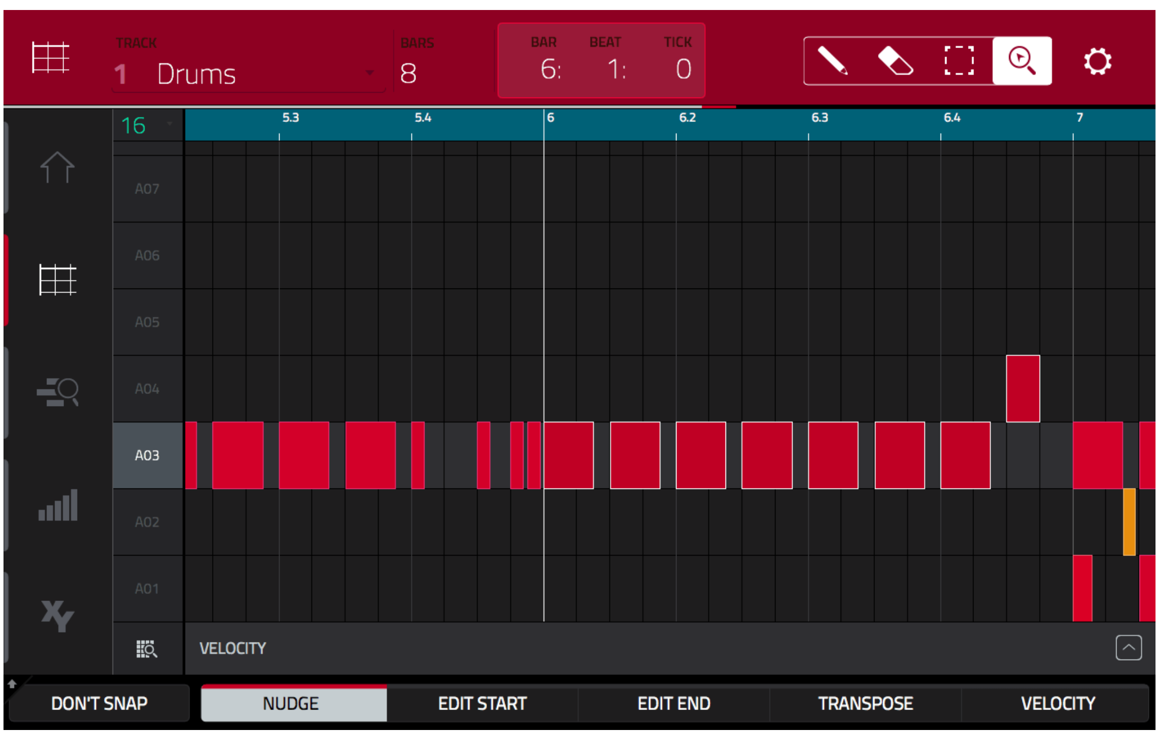


Hit the **COPY** button; this copies the selection to the clipboard. Now, before we paste our hats we nee to prepare the paste location.

First, hit pad **A03** to select the paste ‘row’ to be A03. Now ‘**locate**’ to the **start of BAR 6**, either by using your BAR keys (MPC X) or by tapping the time locator at the top centre of the screen and turning the data wheel until you reach **6: 1: 0**.



Now hold down **SHIFT** again and this time hit the **PASTE** button. The copied hats are pasted exactly where you want them to be, at **006:01:000** on the **A03** row.



Please note*: once you have copied a selection of events to the clipboard they remain there unless you leave GRID EDIT or if you replace them with a new selection of copied events.*

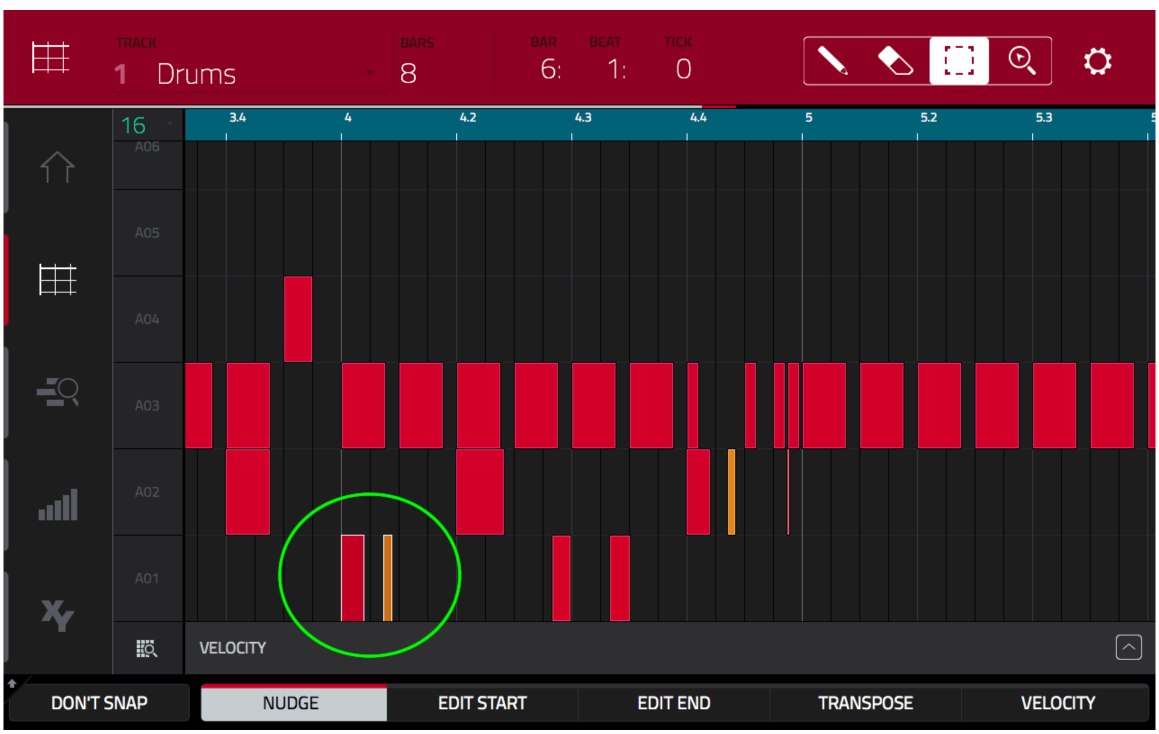
Load up the sequence **Pasted Hats.sqx** for my version.

CUT vs COPY

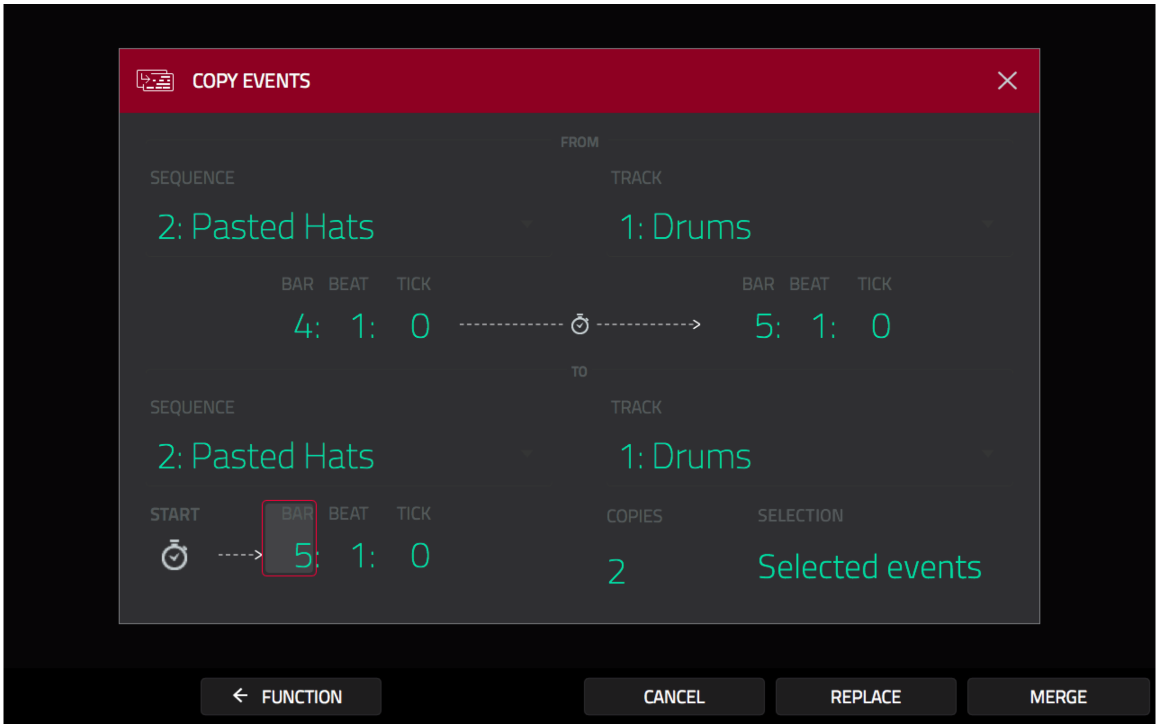
The third edit option in the secondary menu is **CUT** and this can be used to two specific purposes. First it can be used to effectively delete the currently selected notes. But as the ‘cut’ notes are also stored on the clipboard they can then be subsequently pasted anywhere else on the GRID, so here the ‘CUT’ function is effectively acting as a ‘MOVE’ feature.

Copying Events Via MAIN

Let's now copy some kicks. Firstly in **GRID EDIT,** select the kicks we want to copy (this is important) - here I've selected the **first two kicks** at the start of **bar 4** using the select tool:



Now go to **MAIN > pencil icon** (**SEQUENCE row**), select **COPY > EVENTS** and configure as follows:

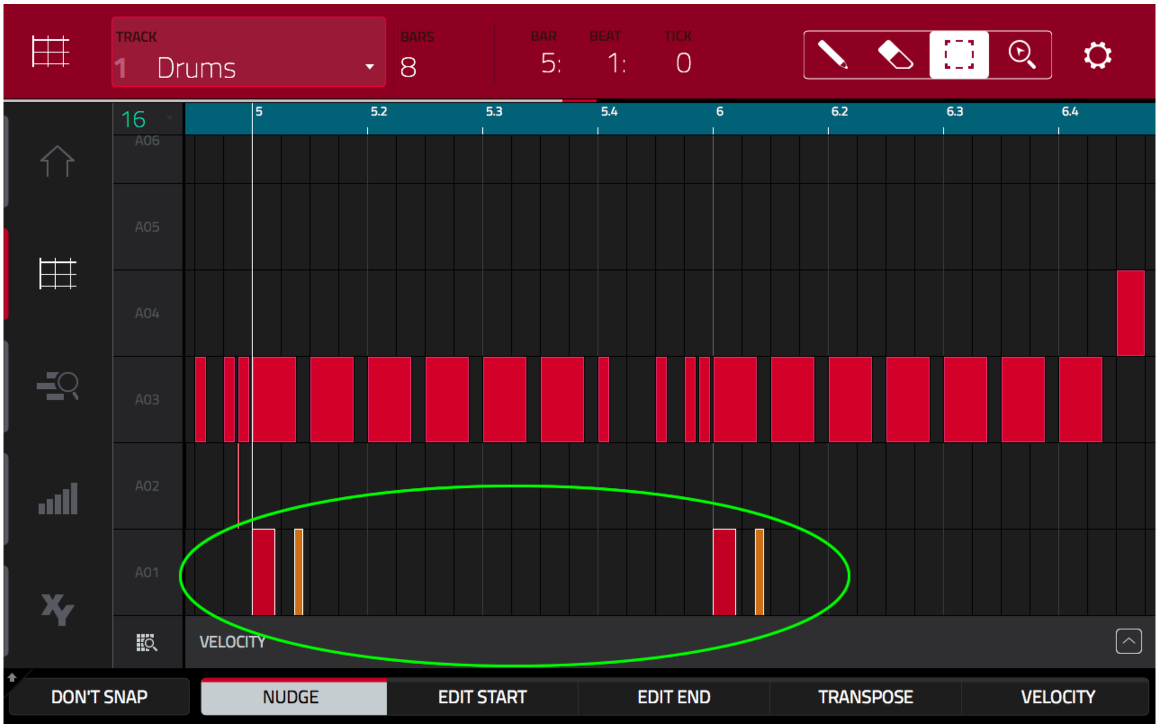


Okay, let's break this down. Under the **FROM** section we've selected our '**Pasted Hats**' sequence, **track 1 (Drums)**. We've set a copy **range** of **4:1:0** to **5:1:0** - i.e. 1 bar.

Under the **TO** section we've set the destination sequence to **Pasted Hats, track 1**, with a paste location of **5:1:0**. We are pasting **2 COPIES**.

Finally notice that I have set set '**SELECTION**' to read '**Selected Events**' rather than 'All Events'. This tells the MPC to only copy the selected events within the 'FROM' range. If we had left this set to 'All Events' it would have copied everything between bar 4 and bar 5, i.e. all hats, snares and all kicks - but now it will just copy the two kick events we previously selected.

Hit **MERGE** to ensure the pasting does not remove any existing events in bars 5 and 6.



The end result is the two selected kick events from bar 4 have been copied and pasted to the beginning of bar 5 and the beginning of bar 6. Load up the sequence **Copy Events.sqx** for my version.

Which copy/paste method do you prefer? If you are copy and pasting within the same track, the GRID method is probably the quickest and most intuitive. For more complex copying across tracks and sequences, the MAIN > COPY EVENTS option is the one to go for.

006 Drum Layering

In this tutorial we're going to spice up our basic drum kit with some very simple but highly effective drum layering techniques!

What is Drum Layering?

Drum layering is the skill of blending together two or more drum sounds to create a completely new sample; as a producer it's your way of defining your own unique drum sound. Drum layering can be as basic as taking a snare and a clap and playing them simultaneously, or it can be a more complex procedure where you carefully bring together many different sounds and textures to build an intricate collage, which can be further tweaked, eq'd and processed.

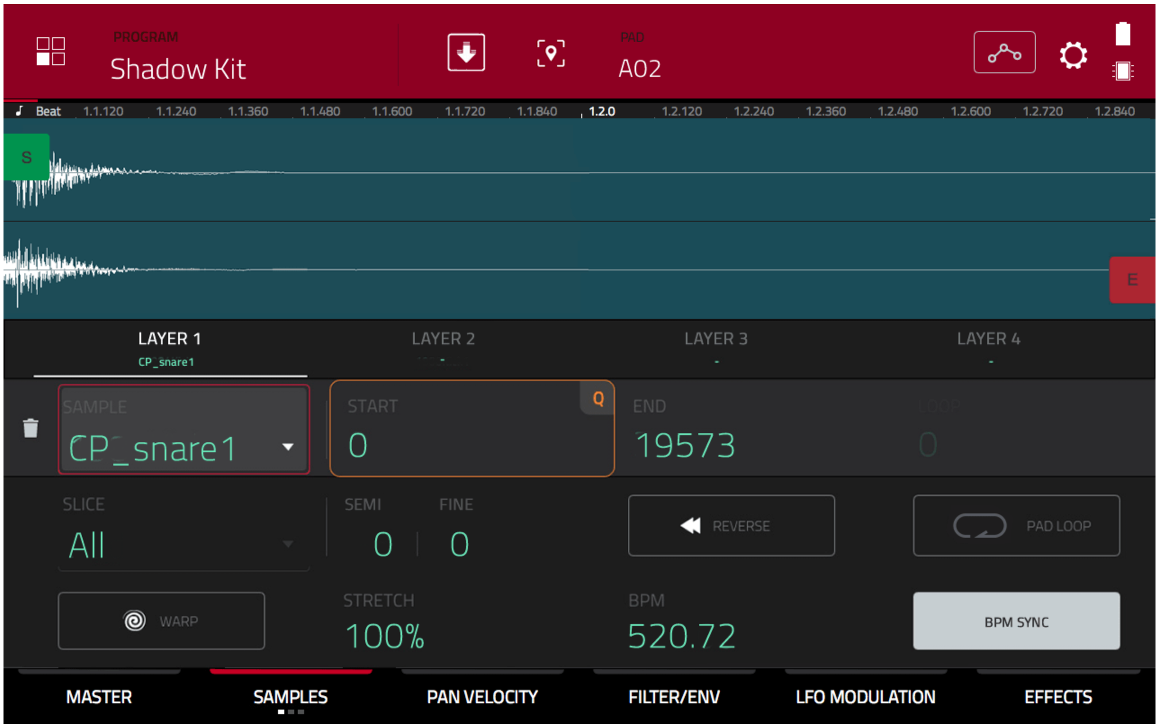
To create our own custom drum layers we'll need to delve into the inner workings of a humble DRUM program. First, head over to the **BROWSER** and enter the **006** folder. Inside you'll find a sub folder called '**Layering Sounds'**. Single tap this folder, hold down **SHIFT** and hit the **LOAD ALL** button at the bottom right of the screen. This will load up a bunch of sounds that we'll use in this chapter for our drum layering experiments.

With the existing **Shadow Kit** program selected in **MAIN**, go to **MENU** and hit the **PROGRAM EDIT** button, or in the MPC X press the dedicated **PROG EDIT** button.

PROGRAM EDIT allows us to adjust various *program parameters* on each pad, which in turn can be used to change the way your assigned samples actually sound (and behave) when a pad is hit.

To organise this mass of parameters, Program Edit is divided into several ‘sub screens’ which can be individually accessed via the touch buttons at the bottom of the screen. The default is the **MASTER** screen, however initially we need to be concerned with the next sub screen, the ‘**SAMPLES**’ screen.

Hit pad **A02** (our snare) and press the **SAMPLES** button at the bottom of the screen:



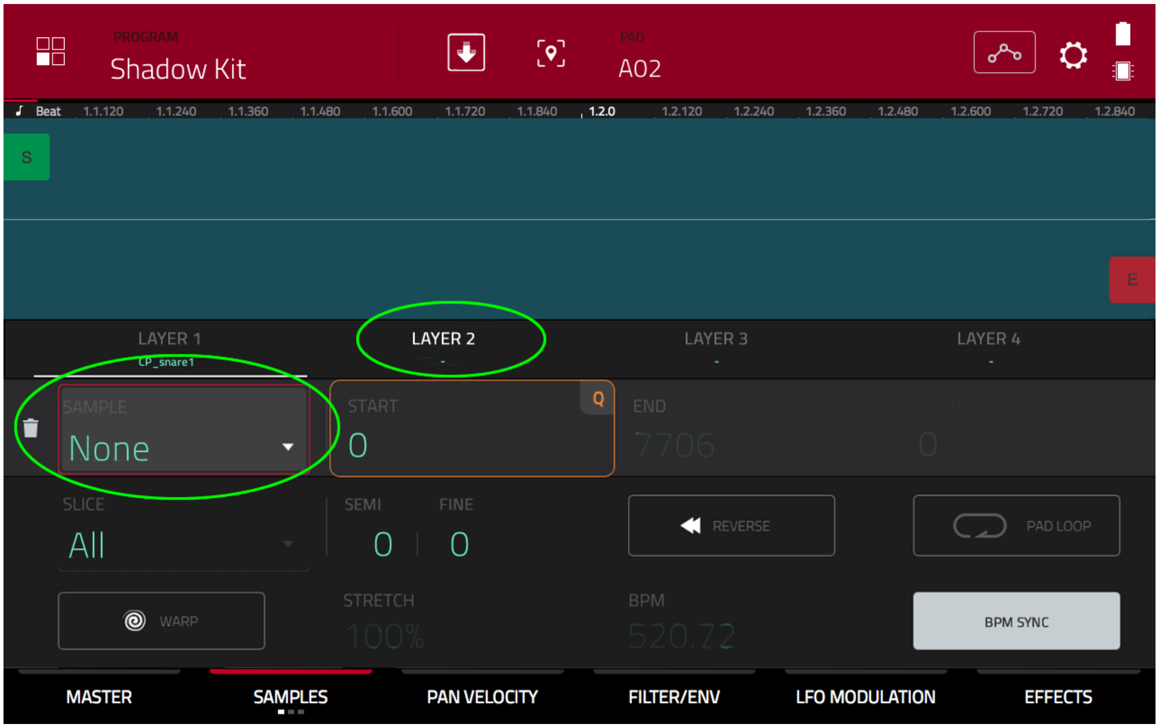
Utilising Sample Layers

The SAMPLES sub-screen handles the **sample layer** information for the currently selected pad. What are sample layers? Well up until now you've just been assigning a *single* sample to a pad, but in a DRUM program each pad is actually able to hold up to **four** samples simultaneously.

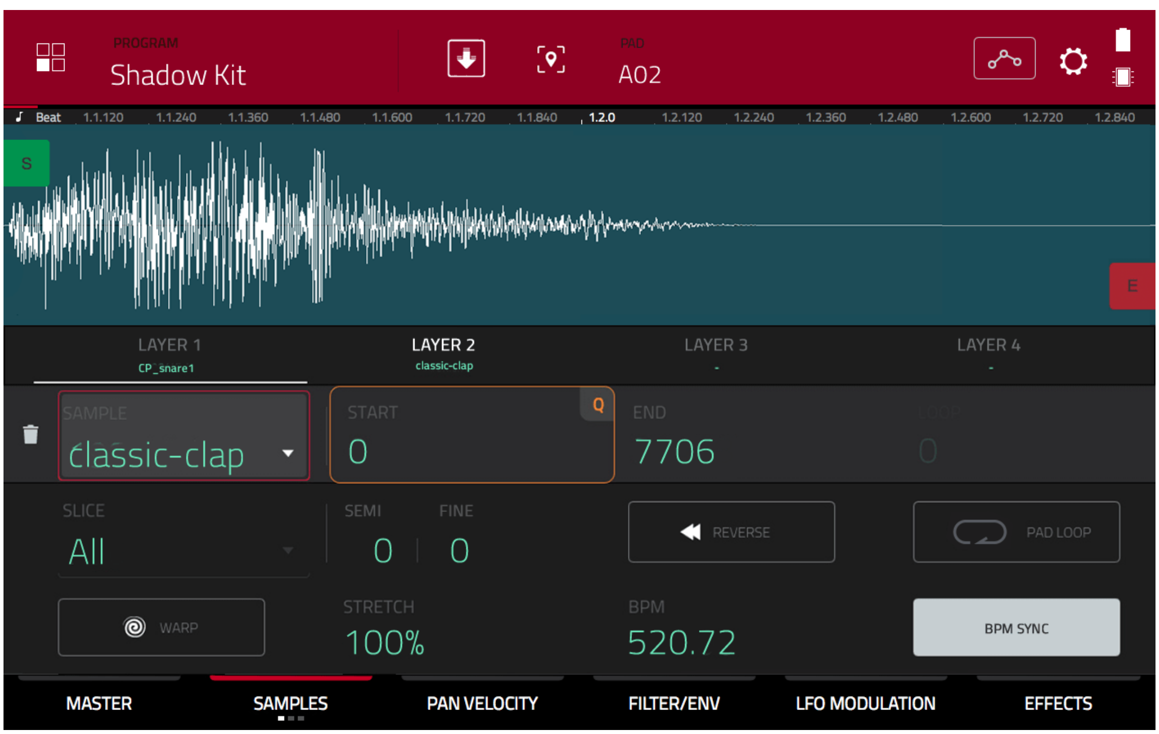
When we assigned a sample to a pad in the previous chapter, we were actually assigning that sample to the 'default' sample layer, 'Layer 1' – 'Layer 1' is simply the first pad layer, which on pad A02 shows the **CP_snare1** sample already assigned from earlier in the book.

The SAMPLES screen contains a lot more information beyond 'layers', such as the sample 'waveform', sample 'edit points', warping and stretching options and much more, most of which is beyond the scope of this chapter, but we'll be looking at all these later in the book.

To assign a sample to the second layer, first single tap **LAYER 2** (centre of the screen) to select the second layer, then tap on the **SAMPLE** box to select it.



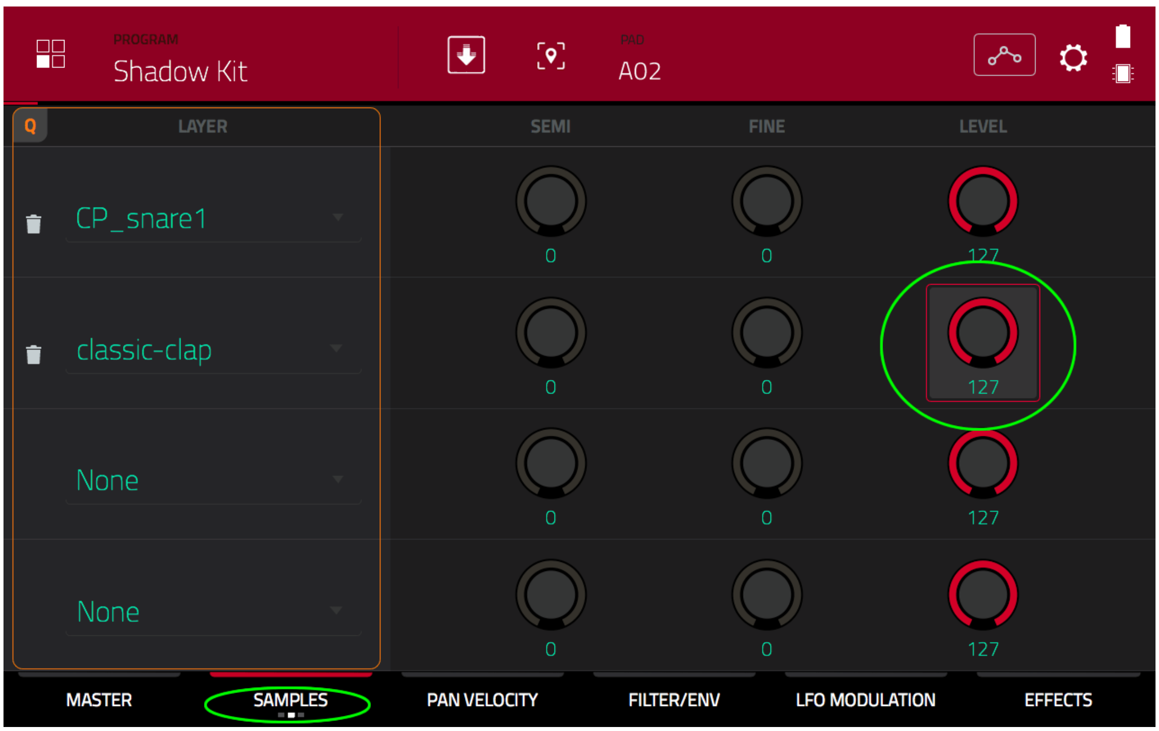
Now turn the data wheel to scroll through all the sounds currently loaded into the Project's sample pool. Choose the '**classic-clap**' sample.



Notice how the trash bin icon appears to the left of the sample name? This will just remove the sample from this layer (i.e. setting it back to '**None**') – it will not delete the actual sample from your Project, so it's still available for assigning whenever it's needed.

Hit pad **A02** to hear our new layered sample. That was pretty easy! However, currently this 'drum layer' doesn't really sound like a snare anymore as the clap layer is much louder than the snare layer. To lower the volume of a layer we need to access SAMPLES 'sub page' 2.

To do this, simply tap on the **SAMPLES** button again, so the **second dot** under 'SAMPLES' turns white:



The second SAMPLES sub page contains additional layer parameters, and it also displays each layer's sample assignments in a more compact fashion on the left hand side.

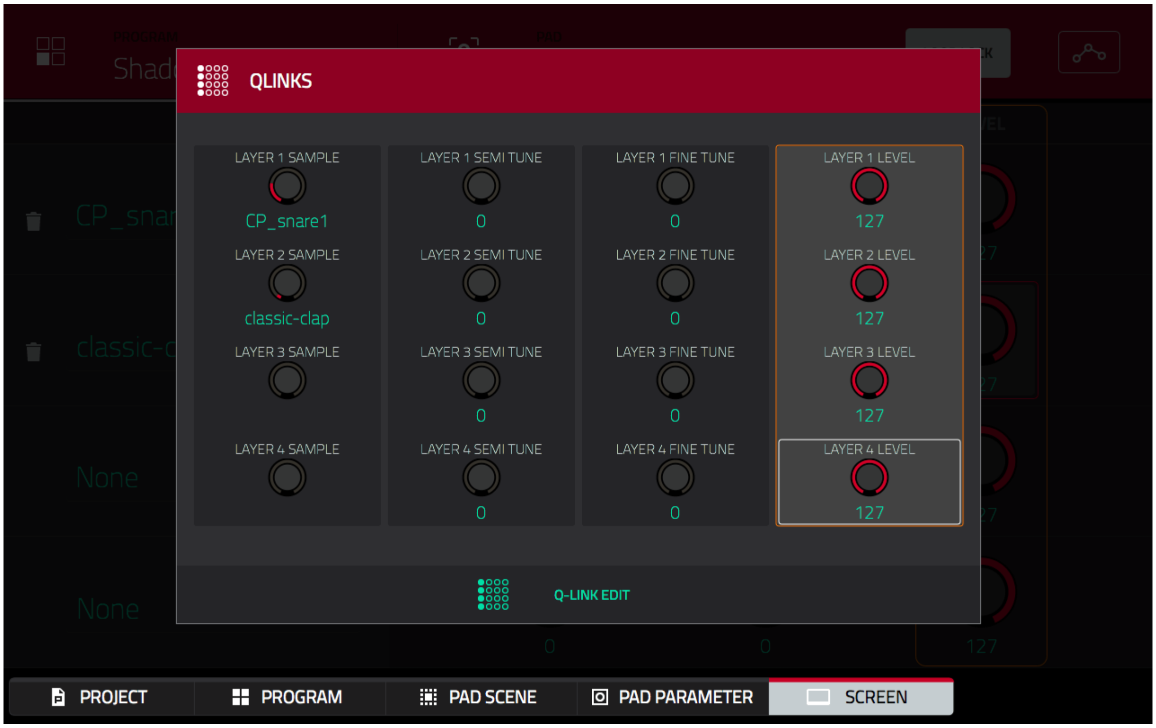
The parameter that controls the volume of each layer is the LEVEL parameter which is found in the last column of this page. There's a few ways we can change the **LEVEL** value. Firstly, we can singe tap and turn the data wheel, secondly we can double tap to bring up an enlarged version of the dial and use our finger to change the value (which I find quite clunky). However at this point I'd like to introduce a third way to perform edits in your MPC; the **Q-LINKS**.

Using Q-Links To Change Parameters

Your MPC has a number of dedicated hardware '**Q-Link'** dials. The MPC X has 16, the MPC Live has four physical dials and three additional columns of 'virtual' Q Links which allow you to select one column of Q-Links at a time.

The Q-Links can be used to change many parameters instead of using the data wheel or touchscreen. By default each mode in the MPC has its own pre-configured Q Link assignments, up to 16 in total on any one page (we'll learn how we can customised these later in the book).

In the MPC X the current Q-Link assignments are shown directly on the OLED display above each Q-Link. In the MPC Live the easiest way to view the page's current Q-LINK assignments is to press and hold the **Q-LINK** button until the **QLINKS** page appears. Here's what it looks like in the **PROGRAM EDIT > SAMPLES** page:



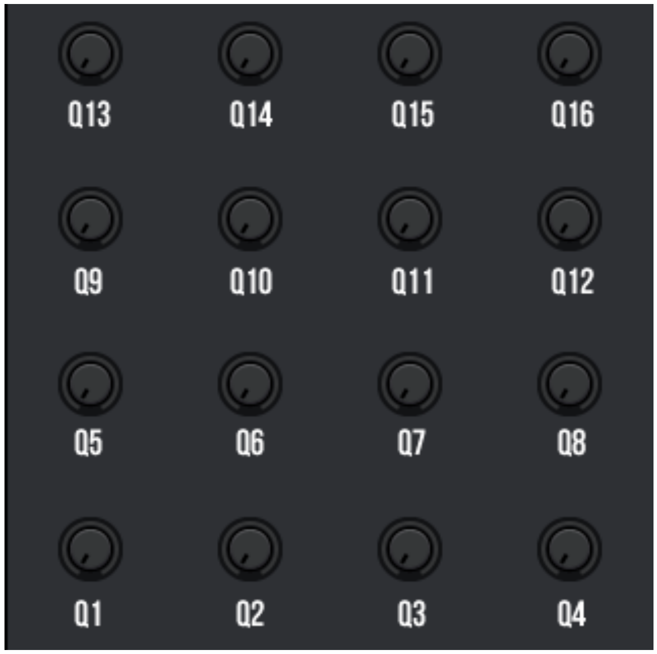
***Please note:** make sure the '**SCREEN**' tab is selected at the bottom on the page - this represents the default Q-LINK settings of the current screen view.*

As you can see, **Q-LINK column 4** is mapped to '**LEVEL**' column for each layer. On the MPC X this relates to the fourth column of Q links dials. As the MPC Live only has a single column of physical Q-Link dials, the fourth column is accessed by pressing the **Q-LINK** button until **the fourth led** is lit.

You should also see an orange perimeter appear around the layers, indicating your dials are mapped to these parameters.

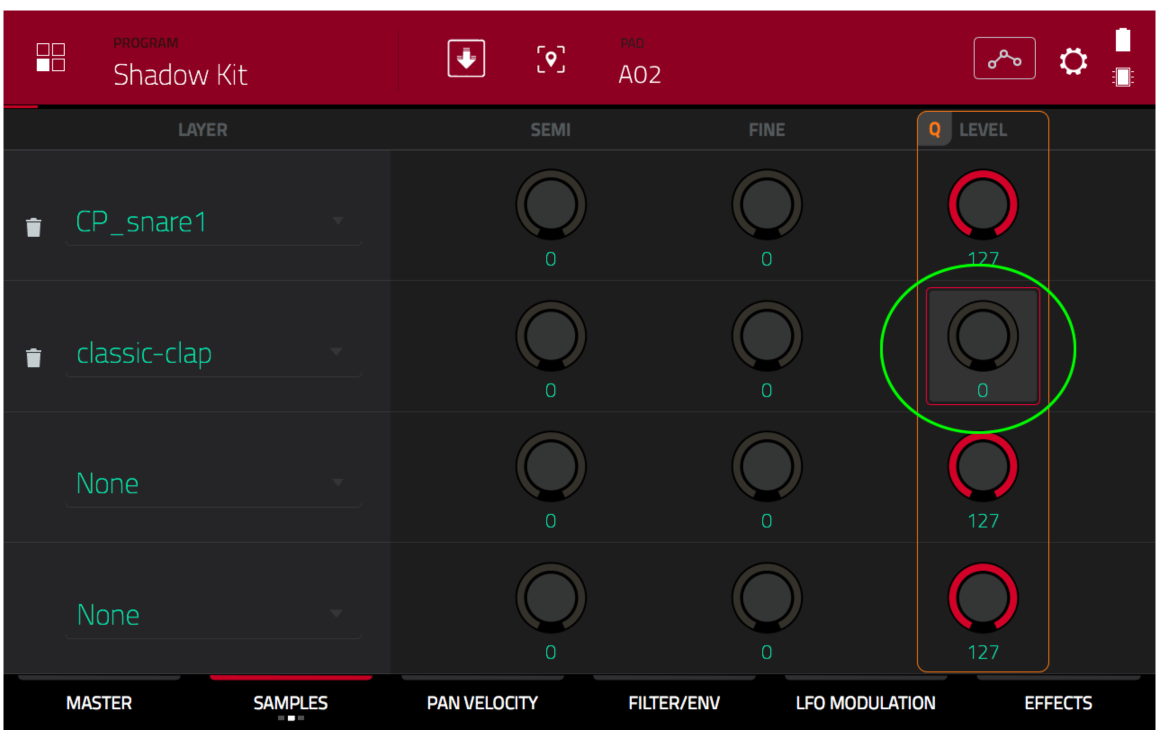
So, to adjust the **LEVEL** value for **layer 2**, we select the **fourth** column Q-LINKs and turn the **second Q-LINK dial** from the top.

We can also refer to the Q-LINKS by their number - now, Akai don't specifically label the Q-LINKS by their number in most software screens nor in the hardware itself, but later in the book we'll use the Q-LINK EDIT page where we find that each Q-LINK dial does actually have a specific number assigned to it:



As you can see, the numbering should be easy to remember as it follows the same numbering as the pads, with Q1 at the bottom left and Q16 at top right.

So using the Q-LNIK numbering system, the LEVEL value for layer 2 (4th column, 2nd Link from the top) relates to **Q12**. So turn **Q12 anticlockwise** until the **LEVEL** for **layer 2** reads '**0**'.

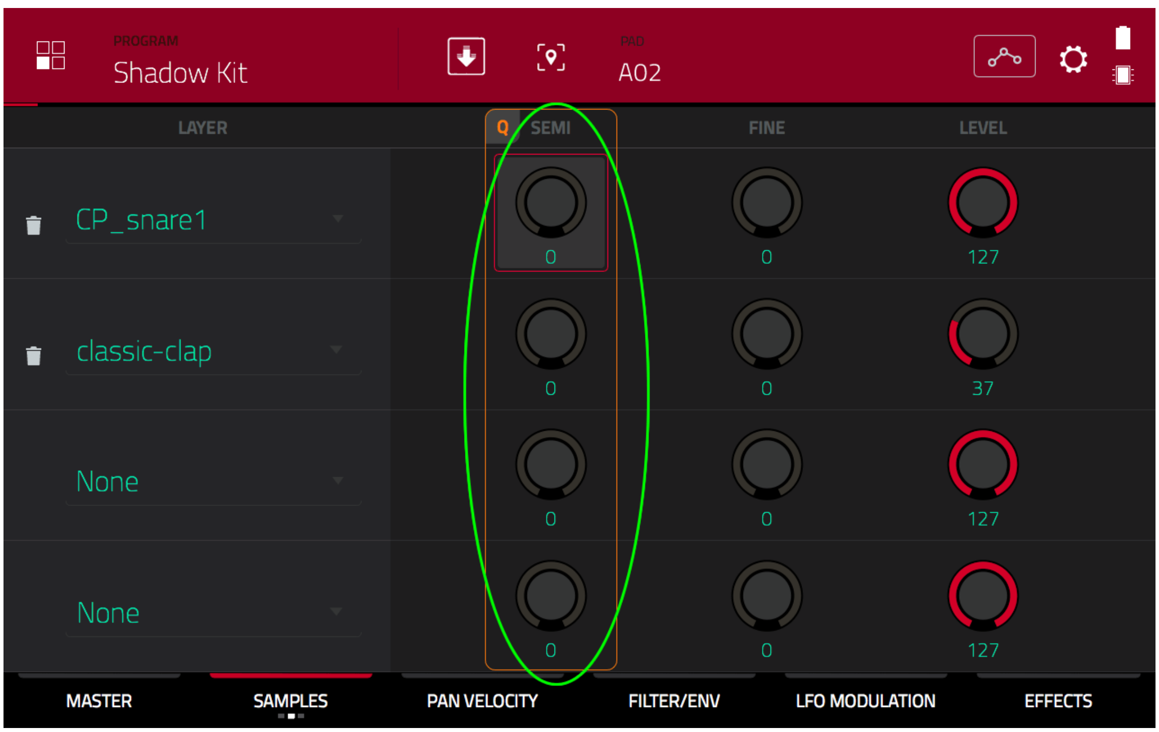


This will completely mute the clap on layer 2. Now start continually tapping pad **A02** and while you do so, begin to increase the **LEVEL** value on **layer 2** so the clap sound is gradually introduced. Stop when you reach a point where you have a nice balance, where the snare has gained a bit of extra body from the layered clap. I set the **LEVEL** on **layer 2** to **37**.

Changing the Pitch of a Drum Layer

We can also adjust the pitch or **tuning** for each layer independently. Tuning is the same as the pitch control slider you find on a turntable. If you increase the tuning, your sample speeds up and raises its pitch. If you tune down (using negative numbers), your sample will slow down and drop in pitch. Tuning can change the tonal character of a drum sound, while on musical sounds it changes the 'key' of the note being played.

Tuning is controlled using column 2 ('**SEMI**') and column 3 ('**FINE**').



To make large changes to the tuning of the clap sample layer, we change the '**SEMI** parameter (**column 2**), which on **layer 2** is controlled by **Q10**, the **2nd from top** Q-Link dial in **Q-Link Column 2**. Turn **Q10 anticlockwise** and while you do, keep previewing **A02** by continually tapping the pad. You'll hear the clap element becomes deeper and darker the lower you drop the pitch down (negative numbers). Set a **SEMI** of **-8**.

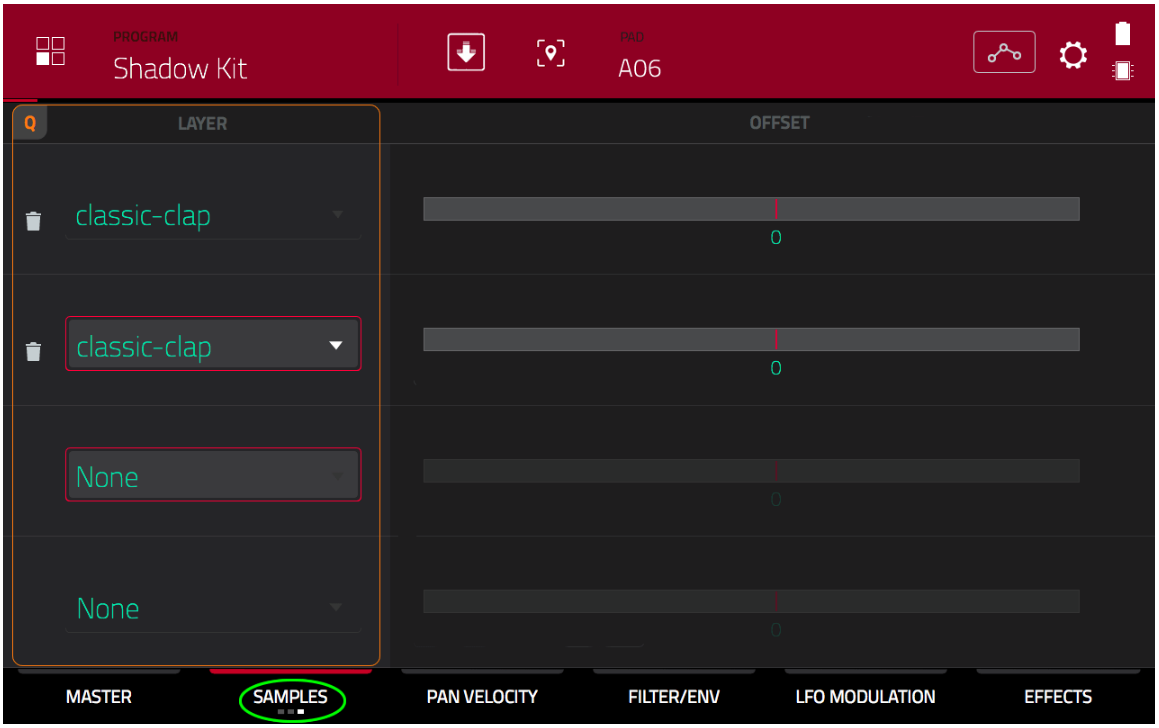
You can also 'fine tune' by adjusting the '**Fine**' parameter via **Q11** (**second from top Q-Link** in **Q-LINK Column 3**). This value goes from **-99** to **99** and this simply allows you to set a fractional tuning. For example, a '**Semi**' of **8** combined with a '**Fine**' of **50** equals an overall tuning of **8.5**. A '**Semi**' of **8** combined with a '**Fine**' of **-50** equals an overall tuning of **7.5**. Set a **FINE** value on **layer 2** to -**50**.

Creating A 'Dragging' Clap Sound

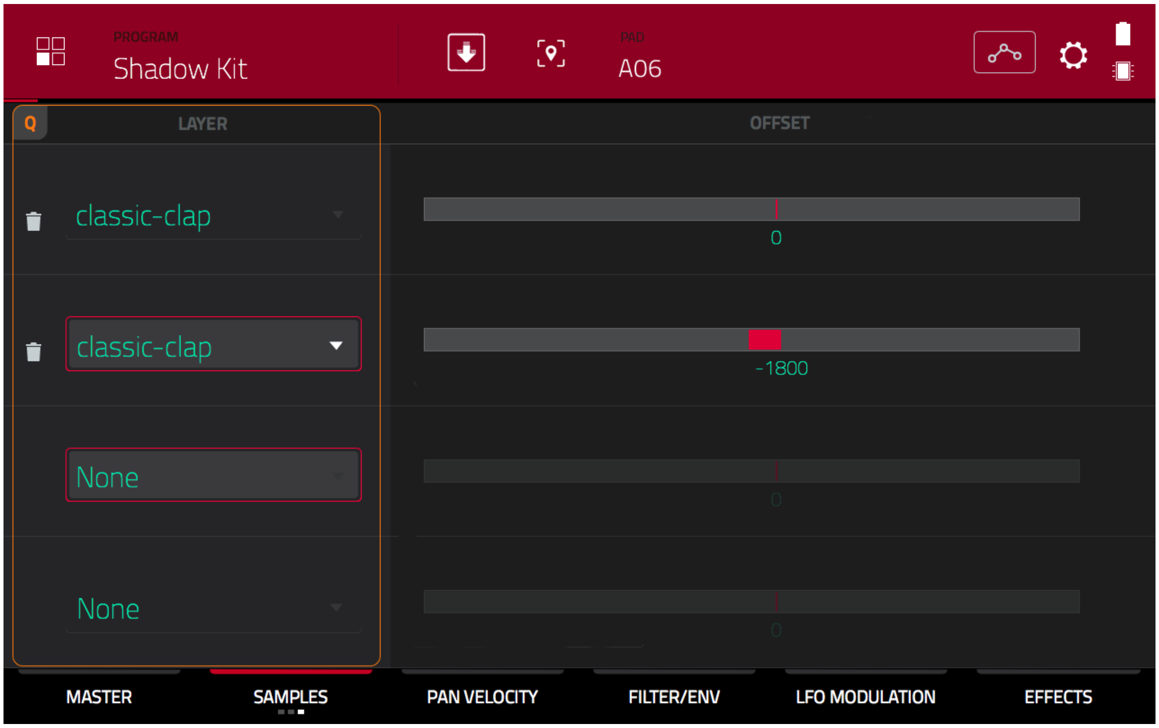
Let's add a standalone clap to our kit. I'm going to use the same clap that we used as the second layer for a snare on pad A02 as this will help the sounds in the kit all sit nicely together.

Hit pad **A06** (which is currently unused) and assign the **classic-clap** sample to both **layer 1** *and* **layer 2**. If you have a listen you'll hear that this clap is pretty loud and a bit 'phased'. Generally it's a bit pointless layering the same sound on top of itself, but in this example it will work really well, as we're going to *offset* the layers from each other. **Layer Offset** allows us to delay the playback of the sample currently assigned to a layer, so when you hit the pad, that layer will not play back immediately

OFFSET parameters are found in the **third SAMPLES sub page**, so press the **SAMPLES** button again so the third dot underneath the button turns white:



What we want to do now is to offset just one of the layers so it ‘comes in’ a little later compared to the other layer. To do this, tap the **OFFSET** parameter for layer 2 and start turning the data wheel anti-clockwise so the offset value is becoming increasingly negative – alternatively use **Q10** (**second Q-Link from the top of Q-Link column 2**). You can also enter a numerical value by double tapping the offset parameter to bring up the number pad.



At lower negative values (e.g. -100), the two layers are just phasing, however as you reach -1200 or greater, the phasing is gone and now we are instead getting a different type of sound. This is the familiar 'dragging' clap sound. Pick a value that sounds good to your ears, I set mine to **-1800**; remember if you are setting values with the dials, hold down the **SHIFT** button to increase the turning accuracy.

Using the Same Sound on Multiple Pads

As you can see you are free to place a sample on multiple pads and even on multiple layers – each 'instance' of that sample acts completely independently to the other instances.

Creating Stereo Width

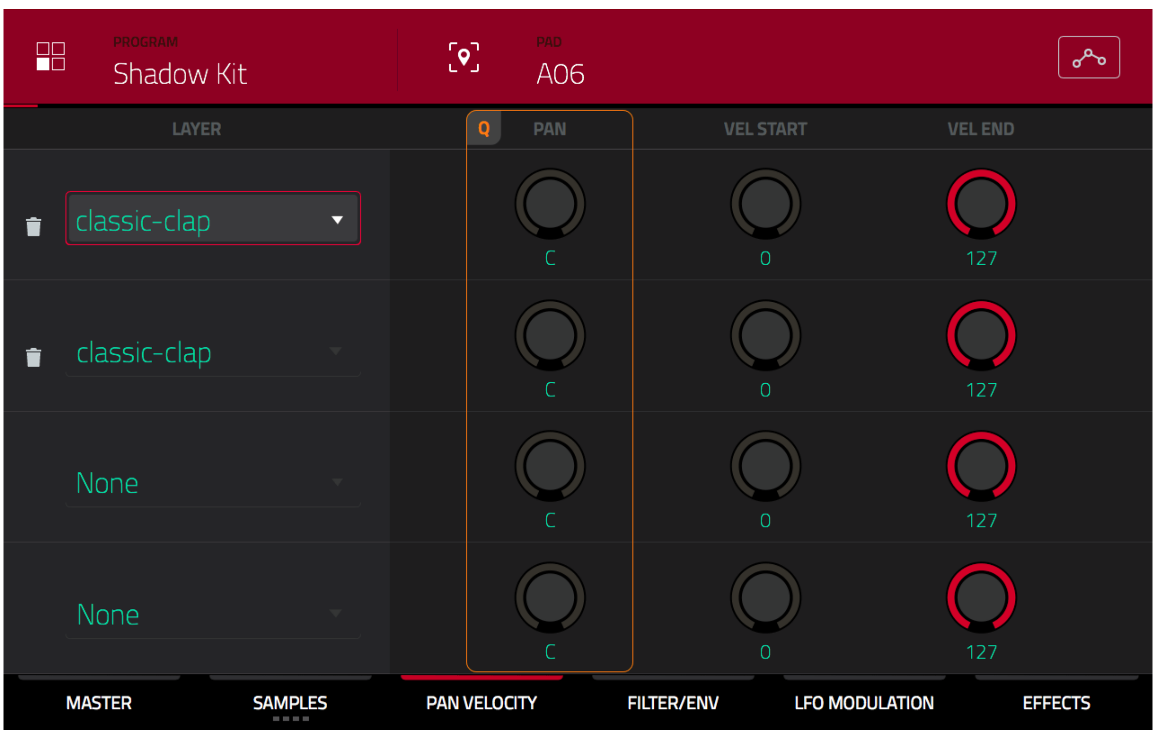
The classic-clap sample is a mono (single channel) sample, but we can make the entire pad even more dynamic sounding by transforming it from mono to stereo.

Stereo vs Mono

A true stereo signal contains two channels of audio (left and right) and gives the listener the benefit of 'panned' sounds, where certain sounds appear to emanate more from either the left or right speakers (headphones will exaggerate this effect). We have two ears and hence hear sounds around us in stereo, so a stereo signal is a way of emulating this. The left and right channels in a stereo signal will always be different. A complete song will normally be in full stereo, as will most drum loops.

A mono signal is only one channel, and hence will have no panning information. A mono sample will still be heard out of both left and right speakers, but the signal in each speaker will be identical.

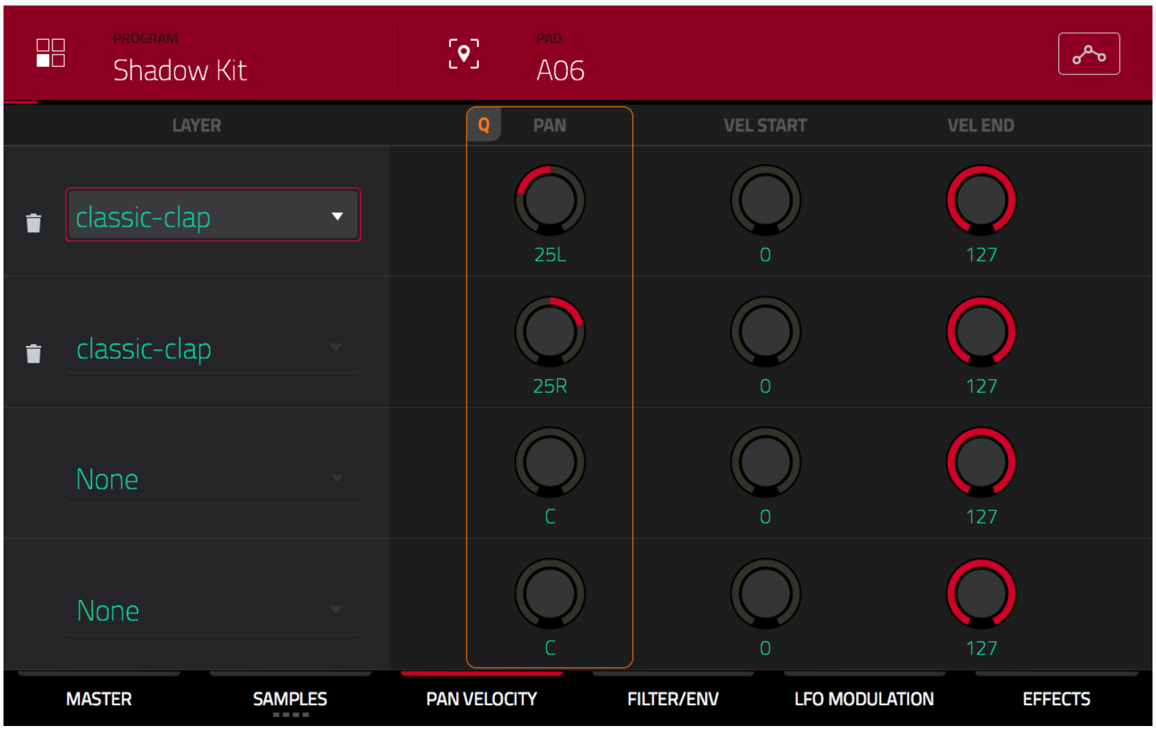
Press the **PAN VELOCITY** button:



The first column here controls the **panning** for each individual sample layer on your pad. Panning controls how much of the outputted sound from your layer is sent to the left and right outputs of your audio interface. With the pan controls at the default '**C**', the output from that layer is sent in equal amounts to the left and right outputs.

Turn **Q13** (the **top Q-Link in column 1)** anticlockwise until the '**Pan'** setting for **layer 1** reads **25L** – this has set the top layer to a left' pan of 25.

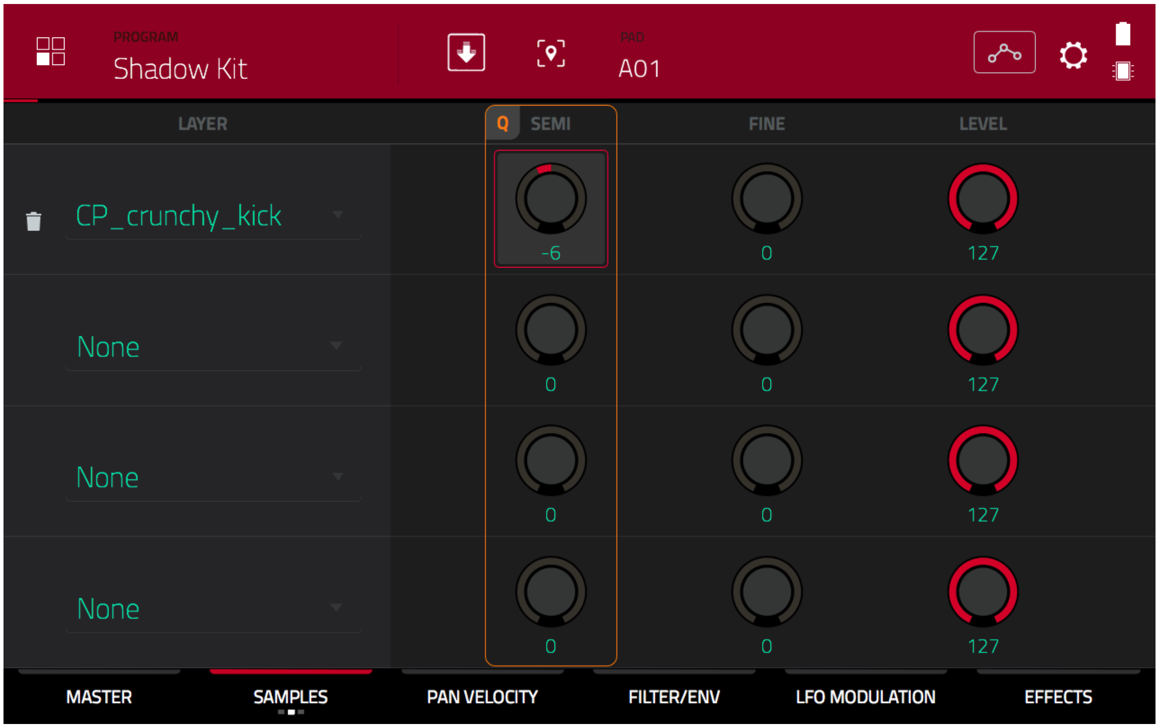
Now start turning **Q9** (the **2nd Q-Link in Q-Link Column 1**) clockwise to gradually start panning the clap on layer 2 to the right hand side. As you do this, keep hitting pad A06 to magically hear how the clap sounds wider. Set the '**Pan'** for **layer 2** to read **25R**.



Creating a Deeper, Crunchier Kick

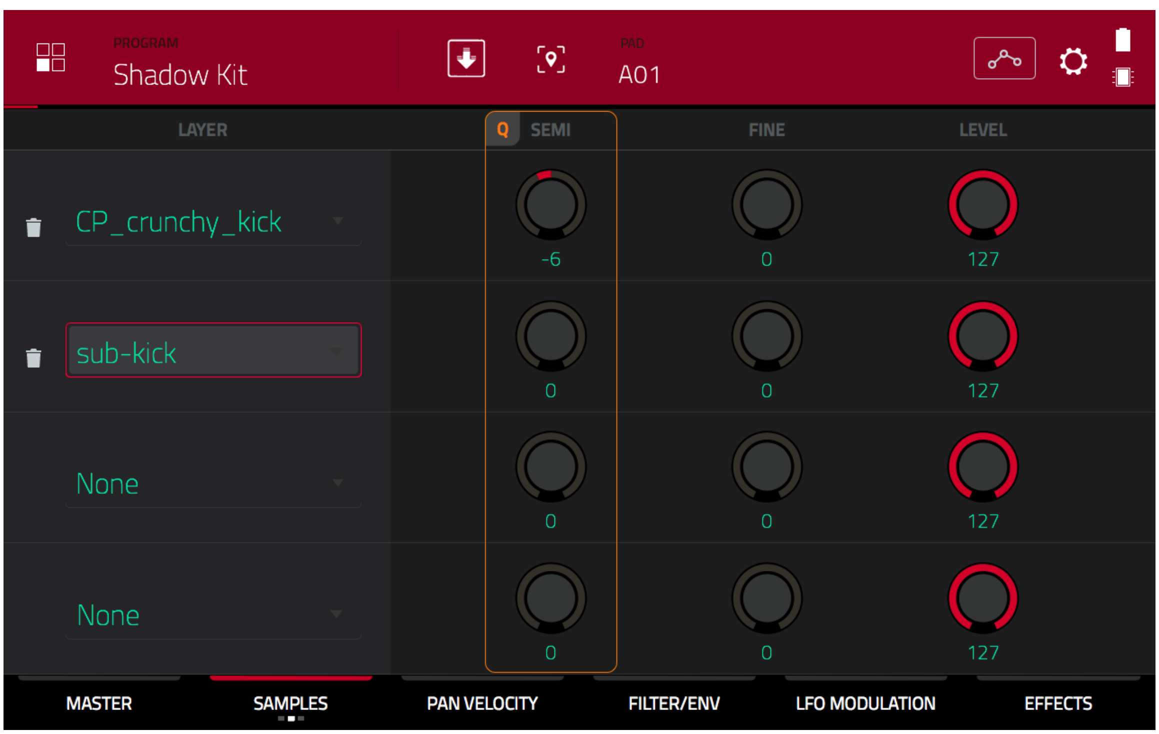
Preview the kick on pad **A01**. Let's face it, it's fairly wimpy, but drum layering will give us the opportunity to create something much fatter and most importantly, a completely unique, 'never-heard-before' kick sound.

Firstly, let's tune this kick down a little. In the second **SAMPLES** screen, take the **SEMI** value for **layer 1** down to **-6**.



Preview the result – the kick already sounds more mean and gritty. Now, how about some more bottom end? Not a problem!

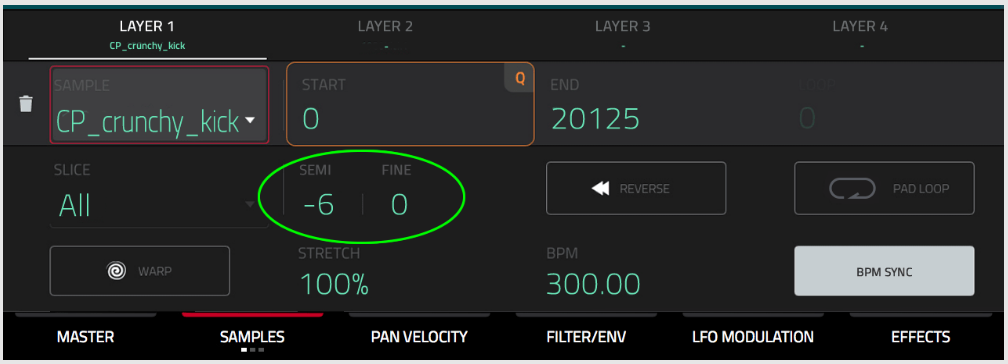
In **layer 2**, select the **sub-kick** sample and preview the resulting drum layer.



That's given our kick a huge amount of sub-bass although perhaps it needs a little bit of taming so the kick sits more comfortably with the rest of the kit. Let's get acquainted with the DRUM program's *envelopes*.

Adjusting Tuning In SAMPLES Screen 1

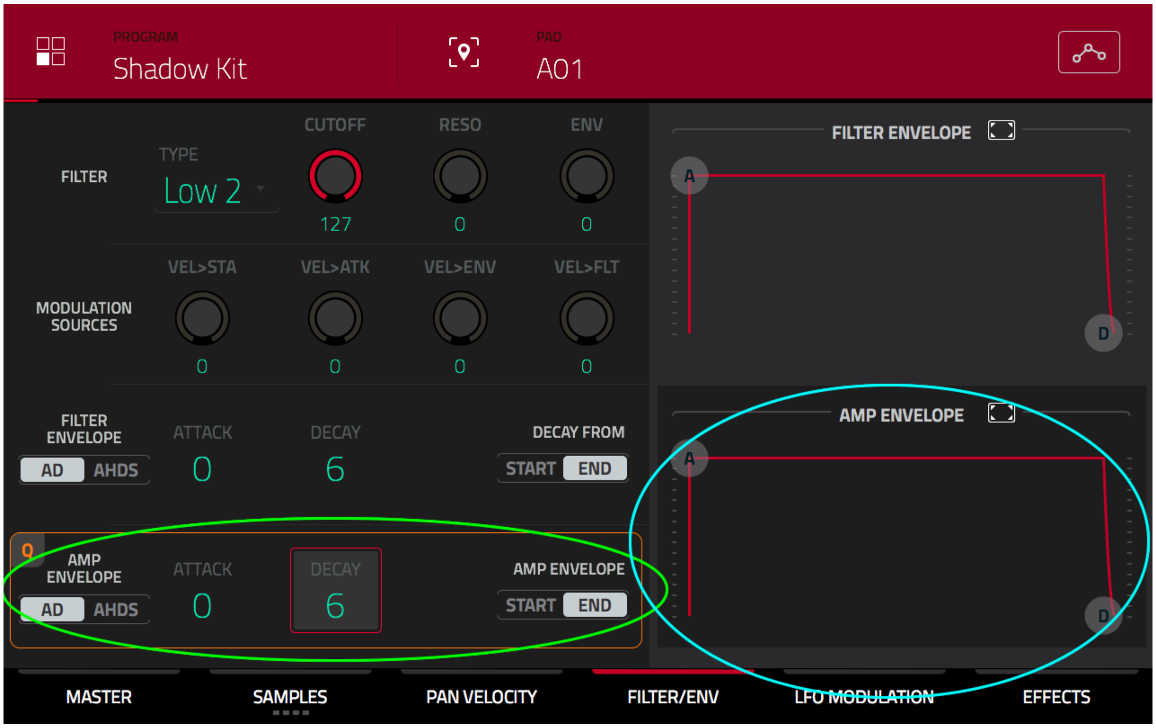
The SEMI and FINE parameters are also available to edit in SAMPLES screen 1 - you can see that our SEMI of -6 on layer 1 is also shown here:



However I prefer the way the second screen shows the tuning and levels for all layers simultaneously, it's just a slightly quicker workflow when tweaking multiple layers together. The settings found on SAMPLES screen 1 are generally more effective when dealing with warping (more on this in later chapters).

Shaping Your Sounds with the Amp Envelope

Press the '**FILTER ENV** button:



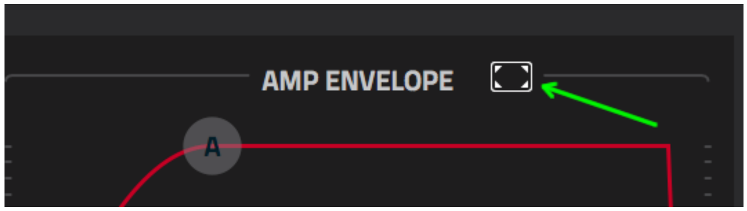
On this sub screen we have the filter and amp envelope settings. The **AMP ENVELOPE** settings (circled in green) allow us to literally 're-shape' the sounds assigned to a particular pad and change the way they play back, as well as changing certain aspects of their character.

The MPC offers a number of different types of Amp Envelope depending on the way a pad is configured in other sub screens, however that's a bit advanced for this section of the book. Here we're just going to take a look at the default Amp Envelope configuration '**AD'**, which is modelled on the traditional legacy MPC Amplitude Envelope.

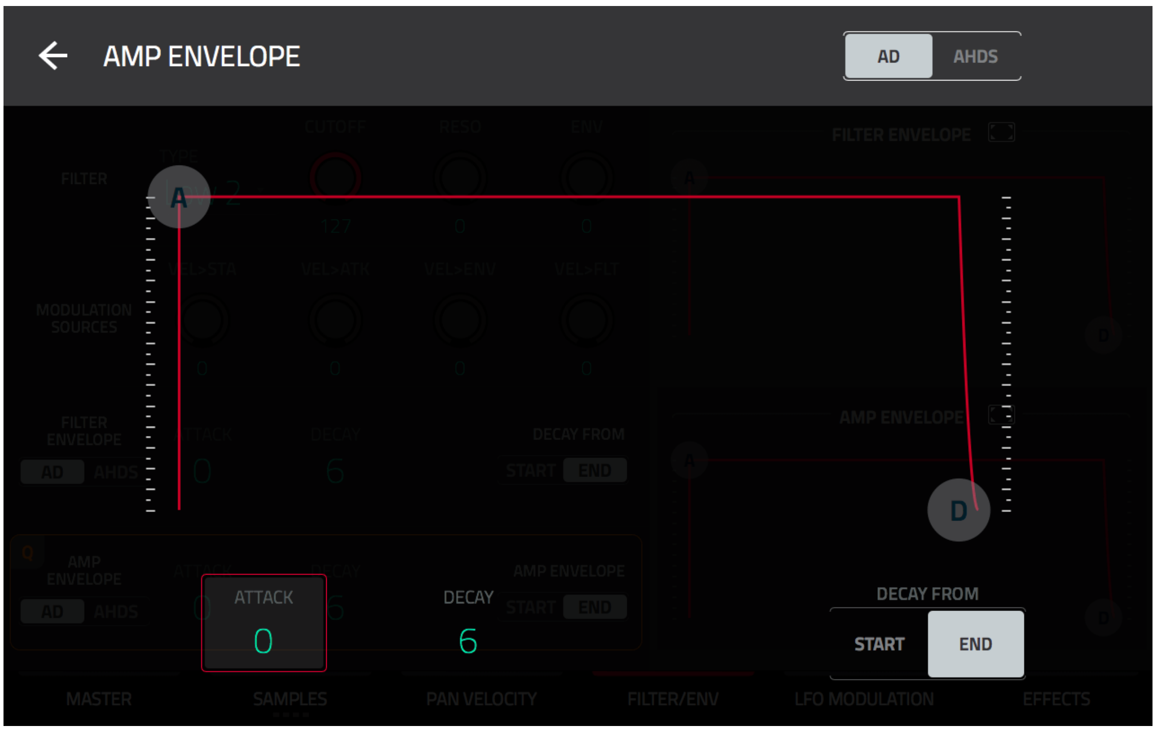
The '**AD'** Amp Envelope consists of two distinct elements; **ATTACK** and **DECAY**. **ATTACK** refers to the shape of the sound at the *beginning* of your sample. ATTACK is a measure of how 'sharp' or 'peaked' the initial sound of your sample is, it basically allows us to 'fade in' the beginning of a sound, while DECAY controls how your sound fades out.

To tame the sub bass length on our layered kick we'll need to adjust the *decay* portion of the envelope. You can adjust the **DECAY** value with the **Second Q Link From the Bottom, Q-Link Bank/Column 1**, or by directly manipulating the graphical AMP ENVELOPE curve (circled in blue), which is handy as it helps you visualise in real time the shape of the AD settings you are applying.

You can see a more optimised view of your AMP ENVELOPE settings by pressing the following icon above the graph:



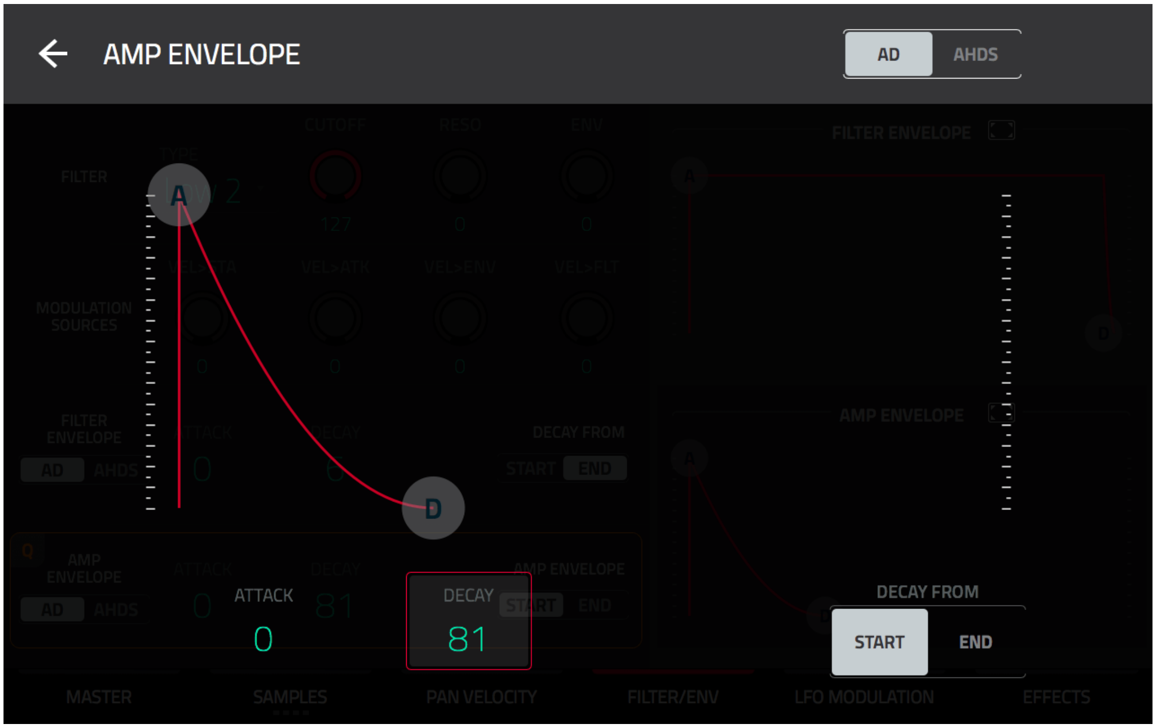
Now only AMP ENVELOPE settings are shown on screen:



Start increasing the ‘**DECAY** value while simultaneously hitting pad A01. You won’t hear much change until you get to a DECAY of around **72**. At this point the sub-bass tail is starting to fade out earlier. Try increasing it a little more until you get a natural sounding, more compact decay. I found **81** worked really well. Take a look at the envelope display now and you can see a visual representation of how the decay is fading our sample:

Hear how the kick sounds shorter and tighter. Now look at the ‘**DECAY FROM**’ type:

The default is ‘**END**’ (**Decay from End**), but change this ‘**START**’ (**Decay from Start**) by touching the ‘**START**’ button. Now the decay envelope acts on our sample from a much earlier point in the waveform, creating a more aggressive and tighter sounding kick.



I actually prefer the kick like this as it keeps it sounding fairly dry, just like the rest of the sounds in our kit, but still retains that deep bottom end.

It's important to note that the envelope settings will change the sound of your *entire* pad, they cannot be used on individual layers (although we'll learn how to work around this limitation in a more advanced tutorial later in the book).

Adding Vinyl Crackle with Simultaneous Play

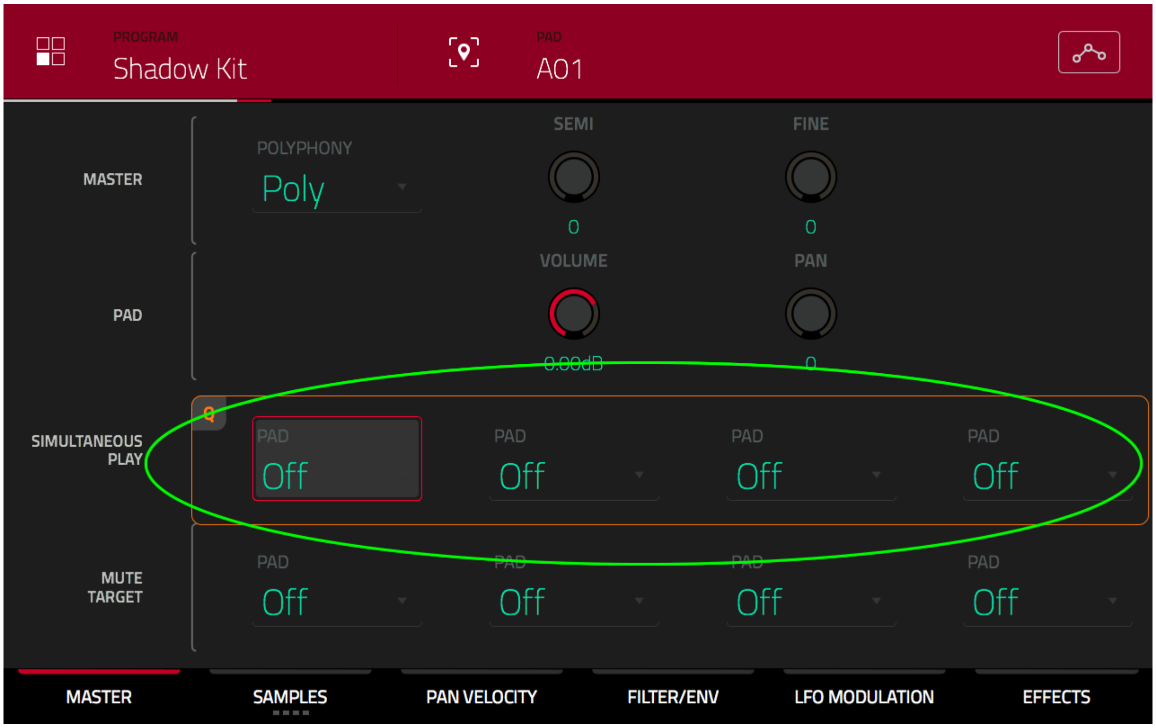
Layering can be used to give your kit a more vinyl sounding flavour by laying some vinyl crackle on top of the sounds.

Now to add a layer of vinyl crackle to a pad we can of course just add a vinyl crackle sample to an empty layer. However the MPC Software has an additional layering option that provides us with a bit more flexibility.

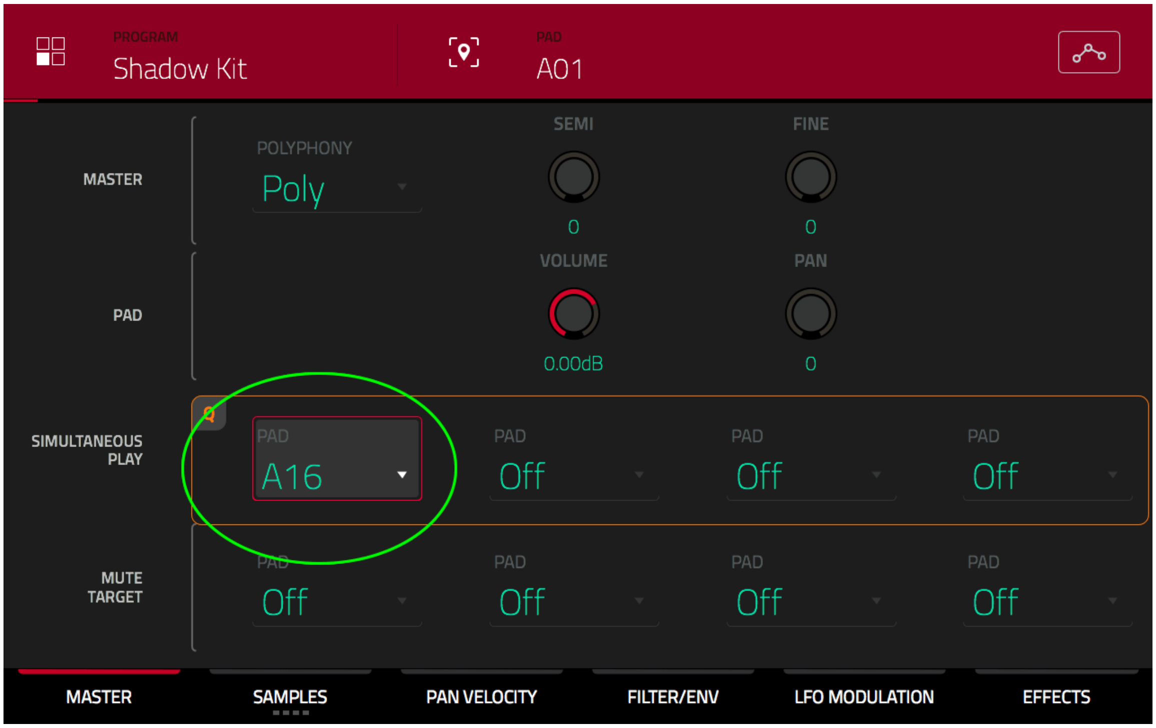
Simultaneous play allows us to trigger (and therefore play back) multiple pads with a single pad hit, so using this method we can effectively 'layer' together entire *pads*. So instead of placing our vinyl crackle on a spare pad layer, we simply place the crackle sample on its own dedicated pad and use simultaneous play to trigger it whenever another pad is struck.

Using this method we can configure our vinyl crackle any way we wish and then layer it simultaneously with any pads we choose.

First let's set up a pad with some vinyl crackle. Hit pad **A16** to select it and in the **SAMPLES** screen assign the **Vinyl-Crackle-1** sample to **layer 1** (hit pad A16 to hear how the crackle sounds). Now select pad **A01** and go to the **MASTER** screen:

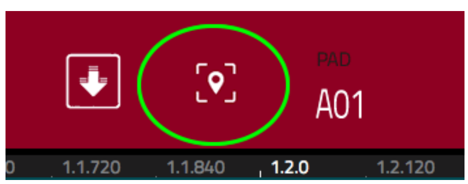


The '**SIMULTANEOUS PLAY**' option is simple to use. On the **SIMULTANEOUS PLAY** line, change the first value from OFF to **A16**.

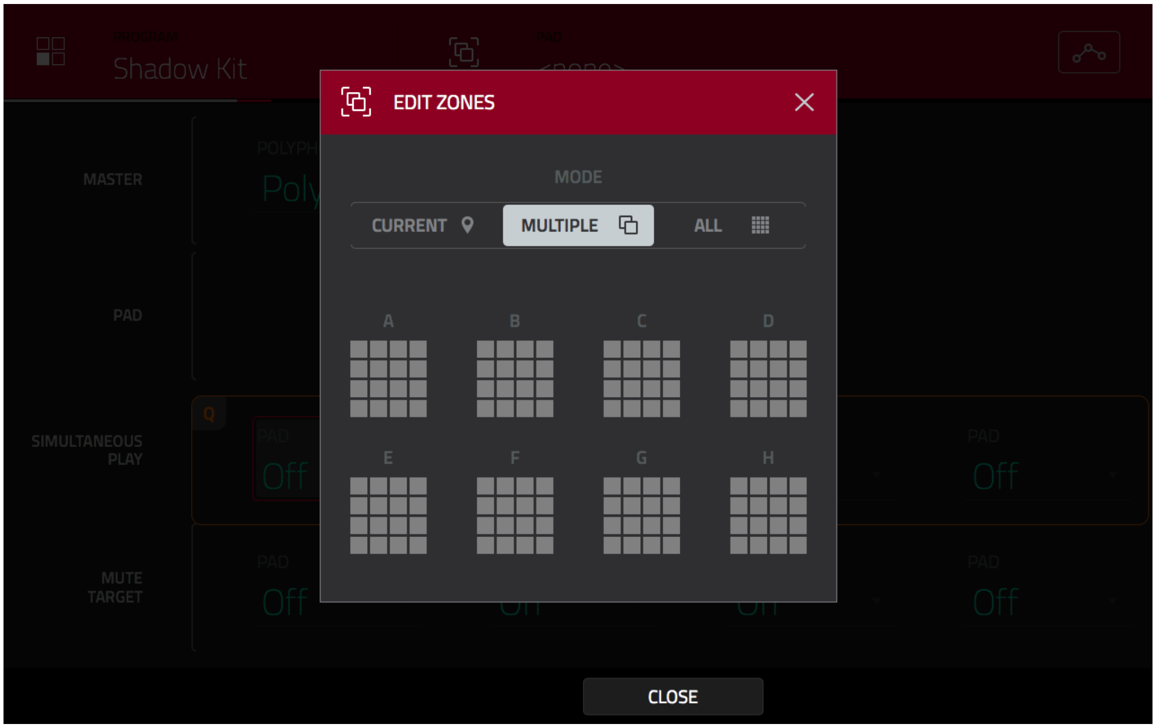


Now trigger **A01** again, and you'll hear the both the kick and the vinyl crackle play back simultaneously - all that is happening is that whenever pad A01 is played, it 'simultaneously' triggers pad A16 as well.

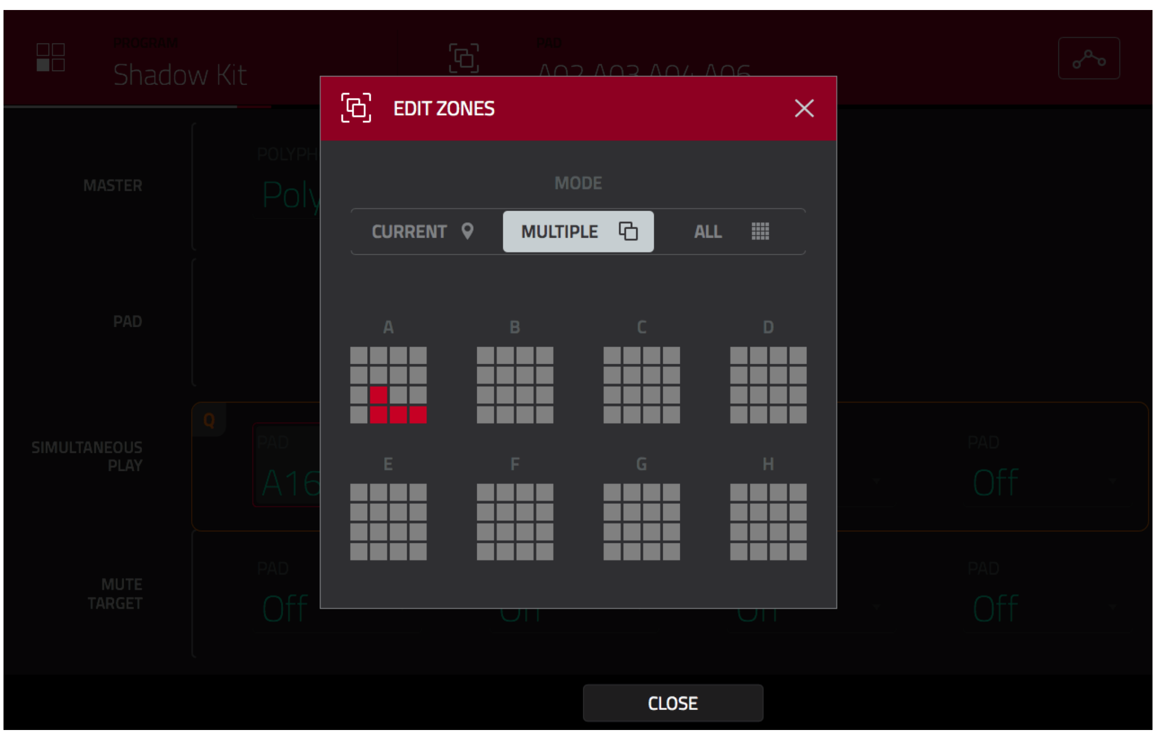
Let's add vinyl crackle to all our pads. Hit the **EDIT ZONES** icon at the top of the screen:



This brings up the **EDIT ZONES** page. Select **MULTIPLE**:



Tap the pads containing drum sounds (you can skip A01 as you already set this); **A02**, **A03**, **A04** and **A06** – this will select all your pads so they can be edited equally together. Selected pads will show in red; to de-select a pad just hit it a second time.



Hit **CLOSE** and hit and **A02**. Set **SIMULTANEOUS PLAY** for **A02** to **A16**. Now hit any of the other selected pads and you'll see that as 'multiple' was selected, all those selected pads now also have A16 added. If you were to change any other program parameter on any one of these pads it would also be set identically on all those pads.

Play your drums and you'll hear they all have crackle playing with them. However it's always the exact same crackle every time which doesn't sound particularly natural. No problem, let's get random!

CAUTION: Before proceeding any further, go back to the **EDIT ZONES** page and set it back to **CURRENT** – this will set the MPC back to standard editing mode.

EDIT ZONES 'SHIFT' Shortcut

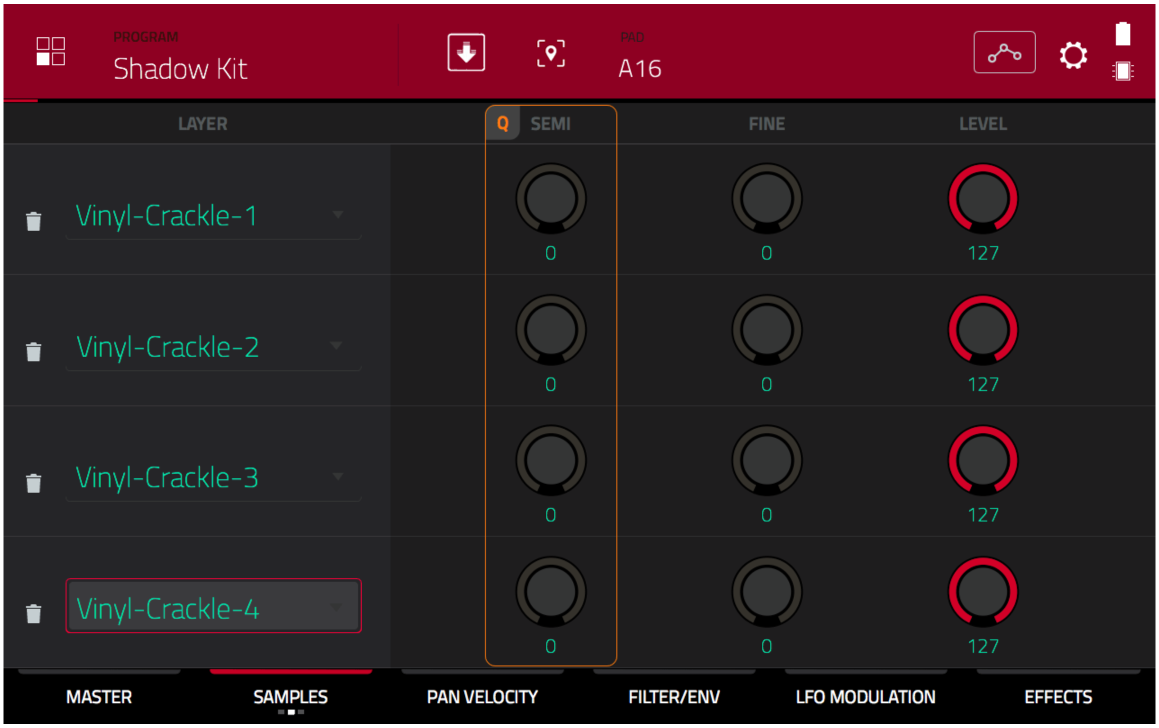
Another way to select multiple pads is to hold down **SHIFT** and touch all the pads you wish to edit together. But you'll still need to return to the EDIT ZONES page to return to the standard '**CURRENT**' mode.

You can also select all pads in a program by selecting **ALL** in the EDIT ZONES page.

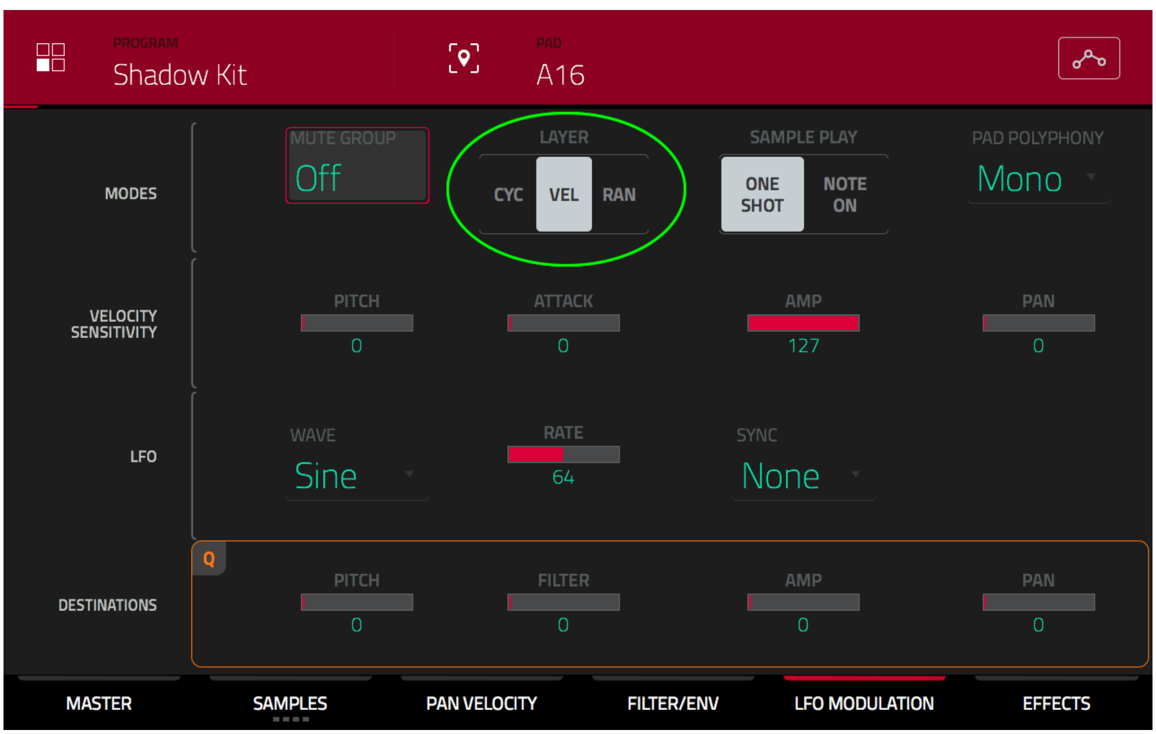
Varying The Vinyl Crackle with Round Robins

Select our vinyl crackle pad, **A16** and go to **SAMPLES** screen 2. Currently we just have a single vinyl crackle sample on this pad, but it's possible to set up a pad to randomly play a different sample *each time the pad is struck*. This functionality is often referred to as a **round robin**.

To set up a round robin we first need to assign our additional vinyl crackle samples to the spare layers on pad A16. From the project sample pool, assign the samples **Vinyl-Crackle-2**, **Vinyl-Crackle-3** and **Vinyl-Crackle-4** to **layers 2**, **3** and **4** respectively.



If you play pad A16 you'll just hear all four crackle samples play together, which is the default playback option for any pad. Go to the **LFO MODULATION** sub screen:



At the top of the page you'll see the **LAYER** option which is currently set to **VEL** (velocity). Change this to **RAN** (random play) and start hitting pad A16 repeatedly. Random play is now randomly playing back only one of the layers each time the pad is hit. Now play the other drum pads and each time you hit one you'll get a different vinyl crackle sound playing in the background giving a more natural sounding vinyl emulation.

You can load up my version of this kit from the **006** folder – **Shadow Kit Layered.xpm**.

CYC Vs RND

An alternative to RAN playback is CYC (cycle). This will play back each layer in strict order (first pad hit will play layer 1, second pad hit plays layer 2 etc). CYC should be used when the order of playback is more important.

In both cases, if a layer is left empty the MPC will simply ignore that layer and just cycle between the used layers.

We'll check out some more advanced drum layering techniques later in the book – we'll also take a look at how we can use the 'VEL' (velocity type of layer playback when we build a multisampled acoustic drum kit. Next up, let's tweak our drum kit even further with some more program parameters and effects.

007 Drum Kit Essentials

Our kit is now really shaping up, but there's a few essential tweaks we can perform to really get this kit set up sweetly!

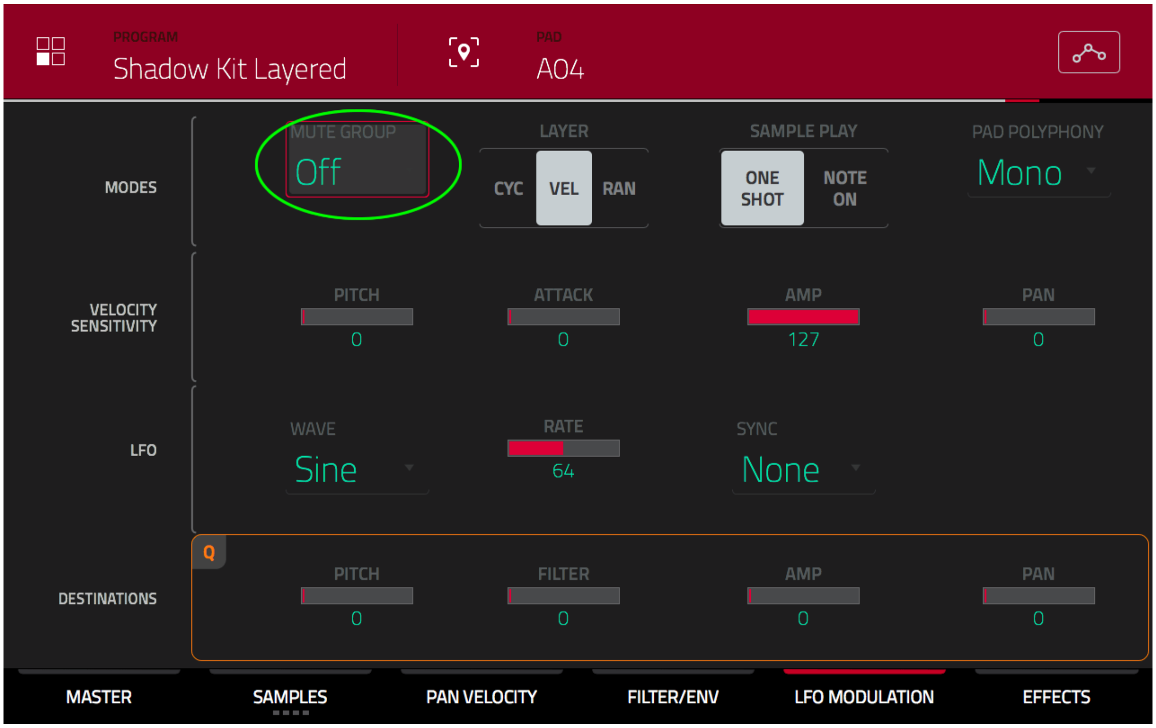
Using Mute Groups For Open Hats

A real hi hat consists of two cymbals that sit on top of each other. When you hit the hat in this 'closed' position, you get the tight 'closed' hi hat sound. When a drummer separates the cymbals using the hi hat pedal and strikes the top cymbal, the two hat cymbals vibrate together to produce the familiar 'open hat' sound.

Select the existing '**Shadow Kit Layered'** program. If you play the open hat on pad **A04** quickly followed by the closed hat on **A03**, you'll hear that the open hat continues to play on top of the closed hat. This does not sound natural because in reality this would not be possible – it's an 'either-or' situation – as soon as you close the hat cymbals together, only a closed hat sound can be made.

To emulate a more natural open to closed hi hat operation we can use '***pad mutes'***. The idea behind a pad mute is simple; the playback of one sample is instantly muted the moment the another specific pad is played. There are actually a few ways this can be accomplished in an MPC, but at this stage I want to look at the ***MUTE GROUP*** method.

Select the open hat on pad **A04** and in **PROGRAM EDIT,** go to **LFO MODULATION**:



The **MUTE GROUP** parameter allows us to assign a pad to any one of 32 *mute groups*. Any pad assigned to the same mute group will always mute each other's' playback, with the most recently played pad always taking playback priority. This is the same concept as the 'choke' groups you find in other samplers such as Maschine and Battery.

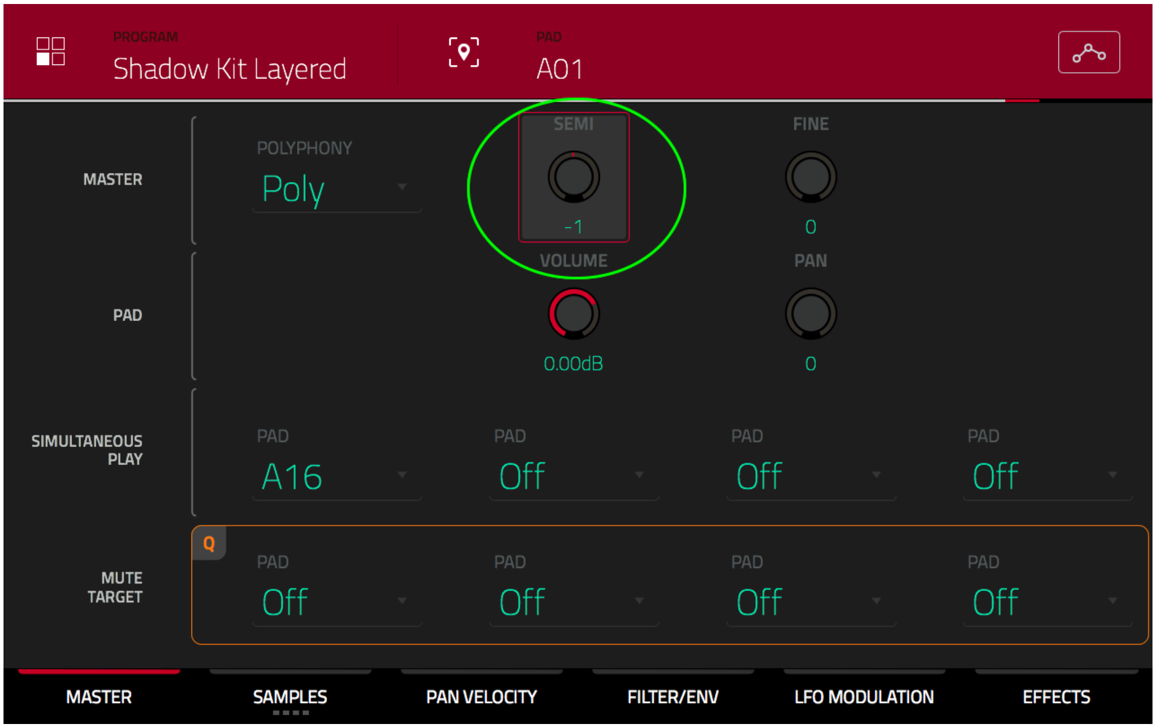
Set the **MUTE GROUP** for pad **A04** to **1**. Now select pad **A03** (the closed hat pad) and set this to mute group **1** as well.

With the mute groups set up, try hitting **A04** followed by **A03** and you'll instantly hear that the closed hat is 'muting' the open hat for you. Suddenly we have a much more realistic sounding hi hat!

Tuning the Whole Kit

We can instantly give the whole kit a slightly darker, grittier vibe by tuning down the whole kit. Rather than tune down each individual layer or pad, we can just adjust the global tuning of our entire program.

Go to **PROGRAM EDIT > MASTER** and tap on the **SEMI** parameter – this will set the tuning for the entire kit. Set this to **-1**:

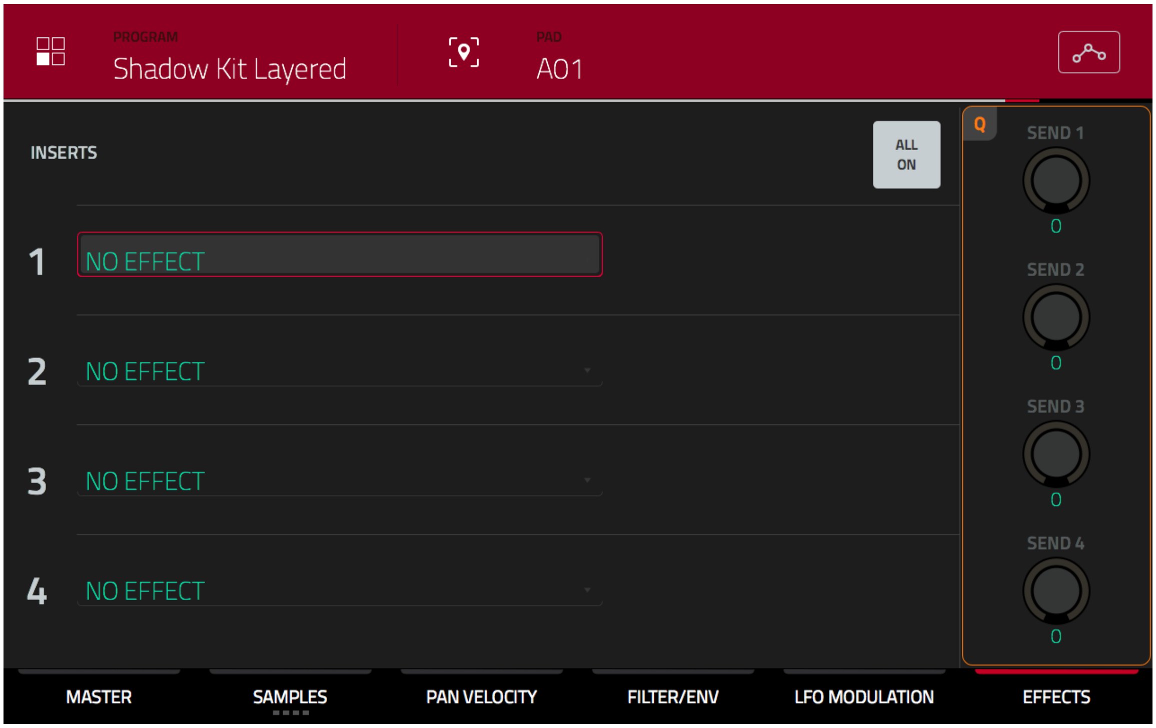


The kit is instantly a little darker. This is a nice subtle tuning, but you can of course tune it even lower, for example a **SEMI** of **-8** sounds like some chilled 90s Trip Hop. You could tune it up to **+9** for a **Jungle/Drum & Bass** style kit.

Adding Some Internal Effects

In addition to program parameters, the MPC ships with a number of built in **effect** plugins provide a means of changing the sonic qualities of the sounds assigned to a pad without making any changes to the actual underlying raw samples themselves.

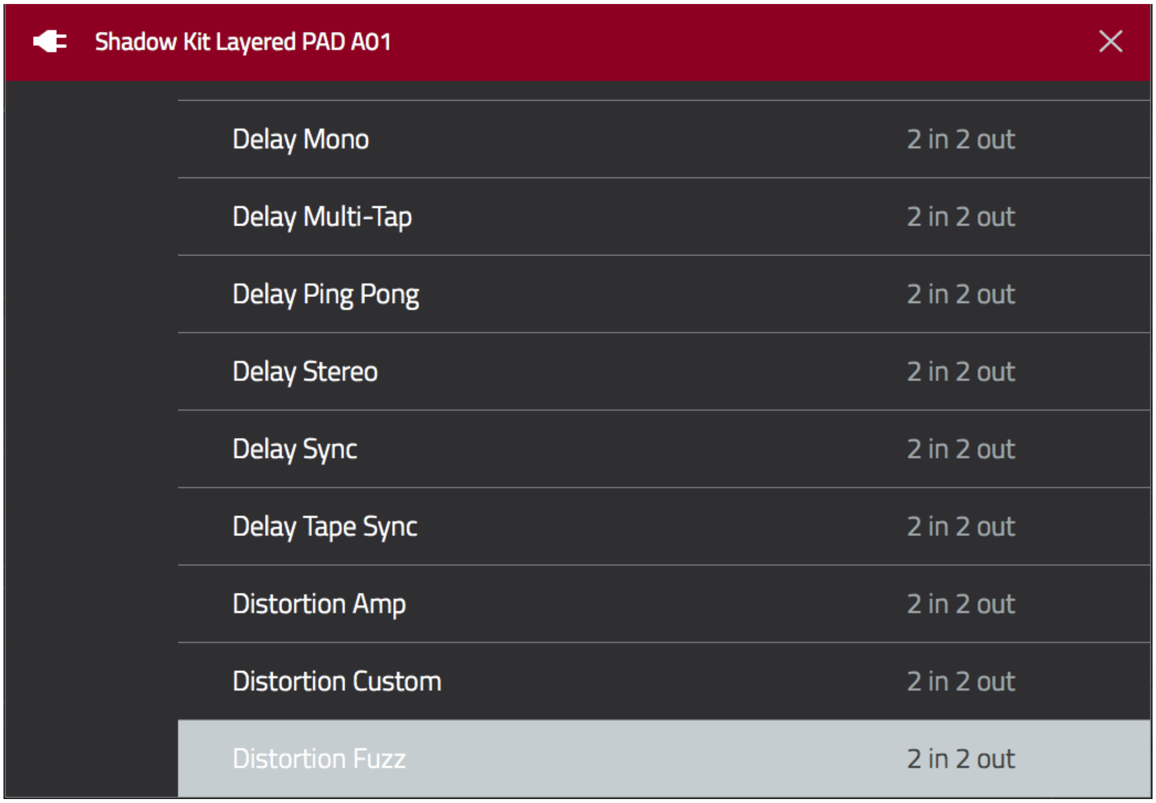
Go to **PROGRAM EDIT** > **EFFECTS**:



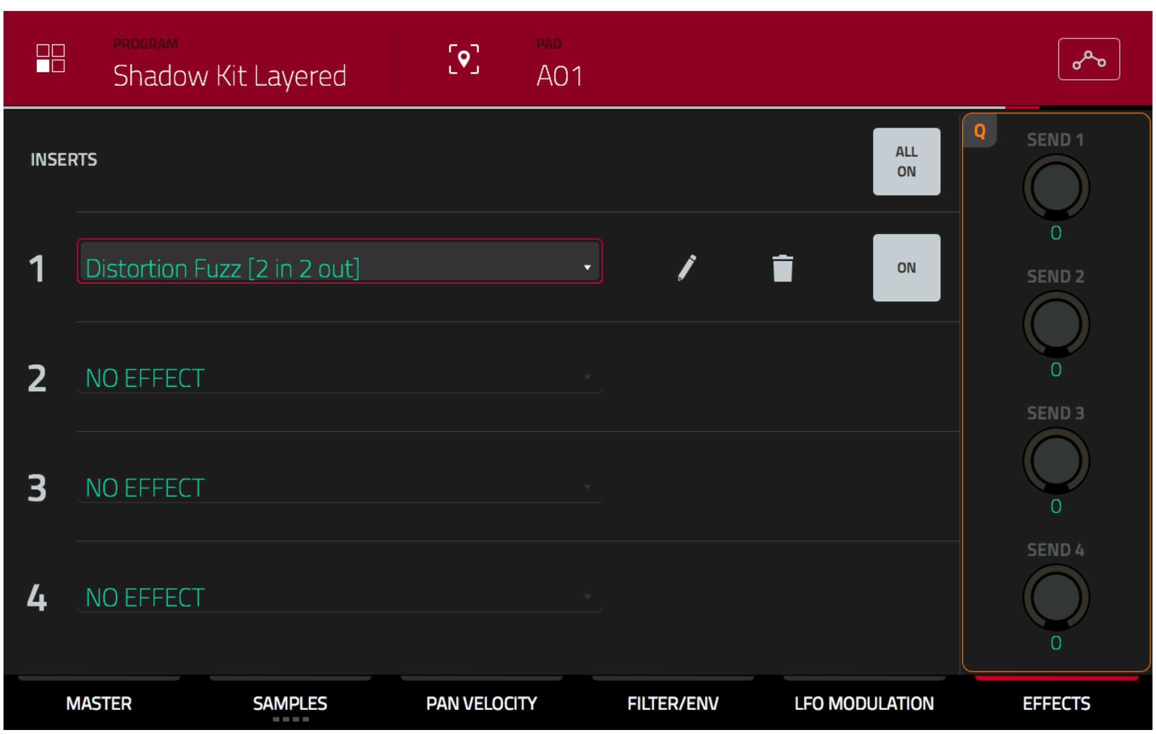
The MPC allows you to insert up to four different effect plugins simultaneously to each individual pad in a DRUM program. The effects section of a program is ideally suited to the use of **insert effects.** Insert effects are used to completely *replace* the original sound with the effected sound, and include distortion, transient shapers, EQ, compression and other dynamics, flanger and chorus.

Applying an insert effect to a pad is incredibly simple. Select the kick on pad **A01** and in the top right of the screen make sure that you set **INSERTS** to '**ALL ON'** – this will activate the effects options for this pad.

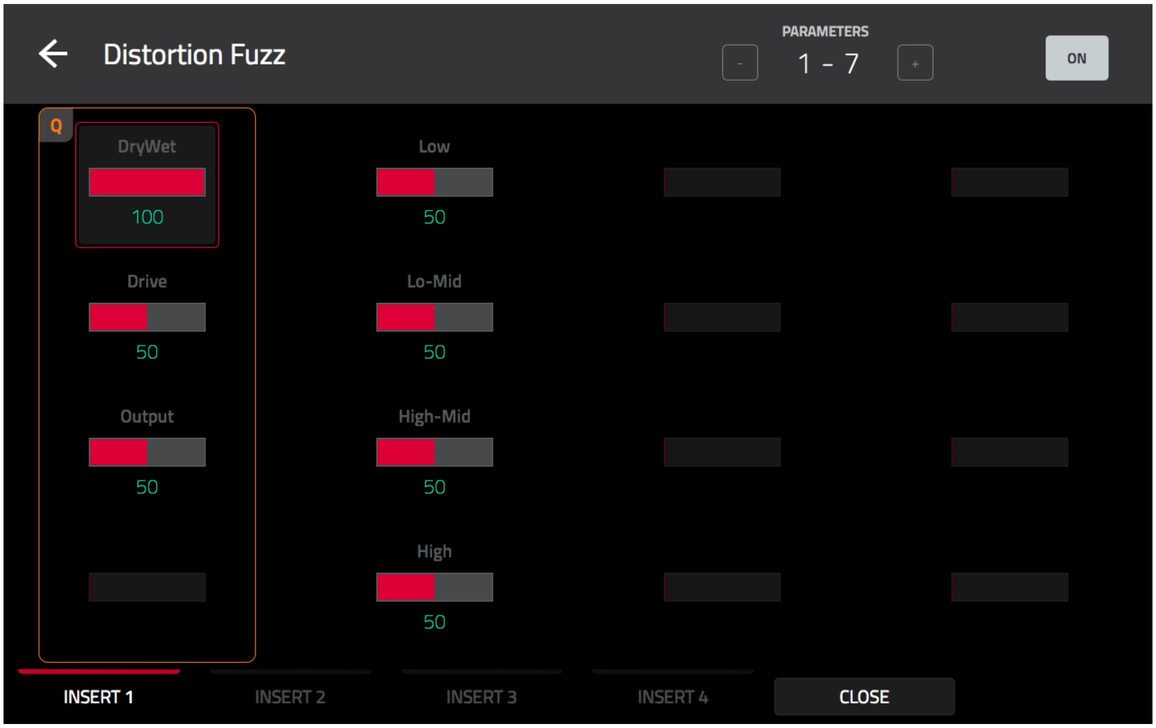
Single tap the **Insert 1** cell and turn the data wheel clockwise to open up the '**Select Effect'** window:



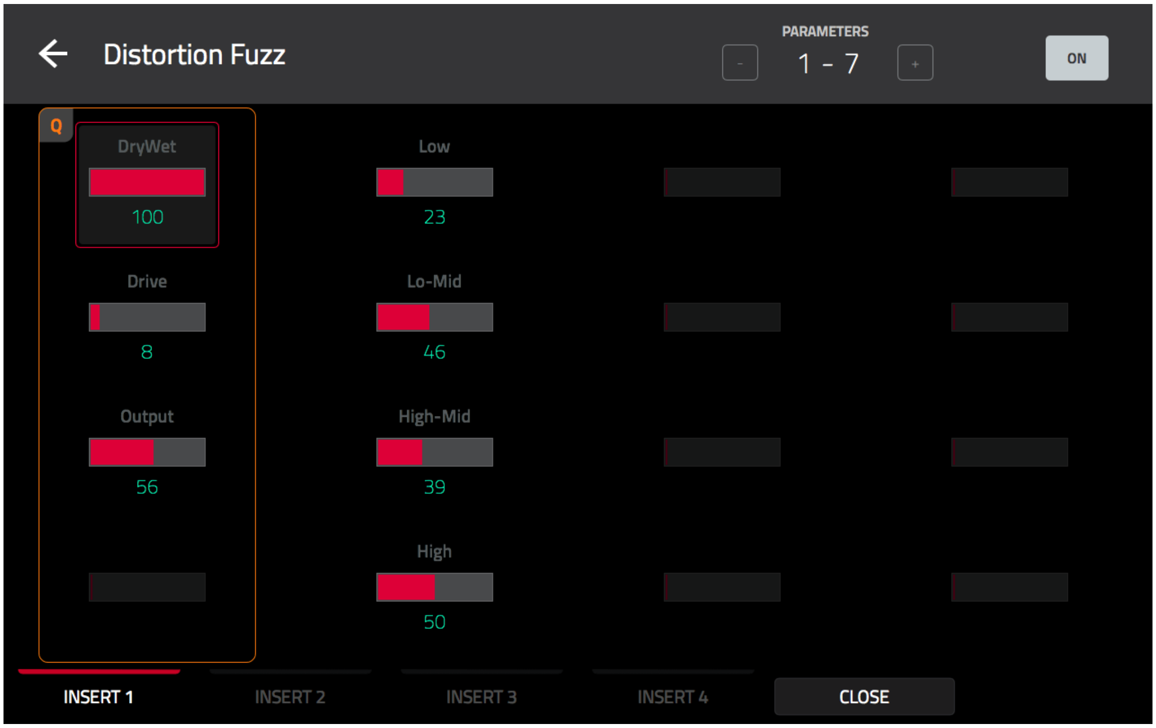
Here you can select any effect from the 'Internal Effects' supplied with your MPC. Scroll until you reach **Distortion Fuzz** and hit **SELECT**.



Preview the kick on **A01** now. It sounds very different! To edit the effect parameters, click on the **pencil icon** in **INSERT 1**.



This is a pure 'Insert' effect, so ideally I will leave the **DryWet** parameter set to **100** (i.e. only the effected signal is heard). It's clear the default settings are distorting the sub bass elements of our kick layer, but perhaps a bit too much. To reduce this, we can lower the **DRIVE** setting, but we can also adjust the **LOW** setting – this lets us tweak the frequencies at which the distortion acts. You can also adjust the **OUTPUT** to compensate for any changes in volume from adding distortion. Don't go overboard with this, it's supposed to be subtle; I set the **DRIVE** at **8.**



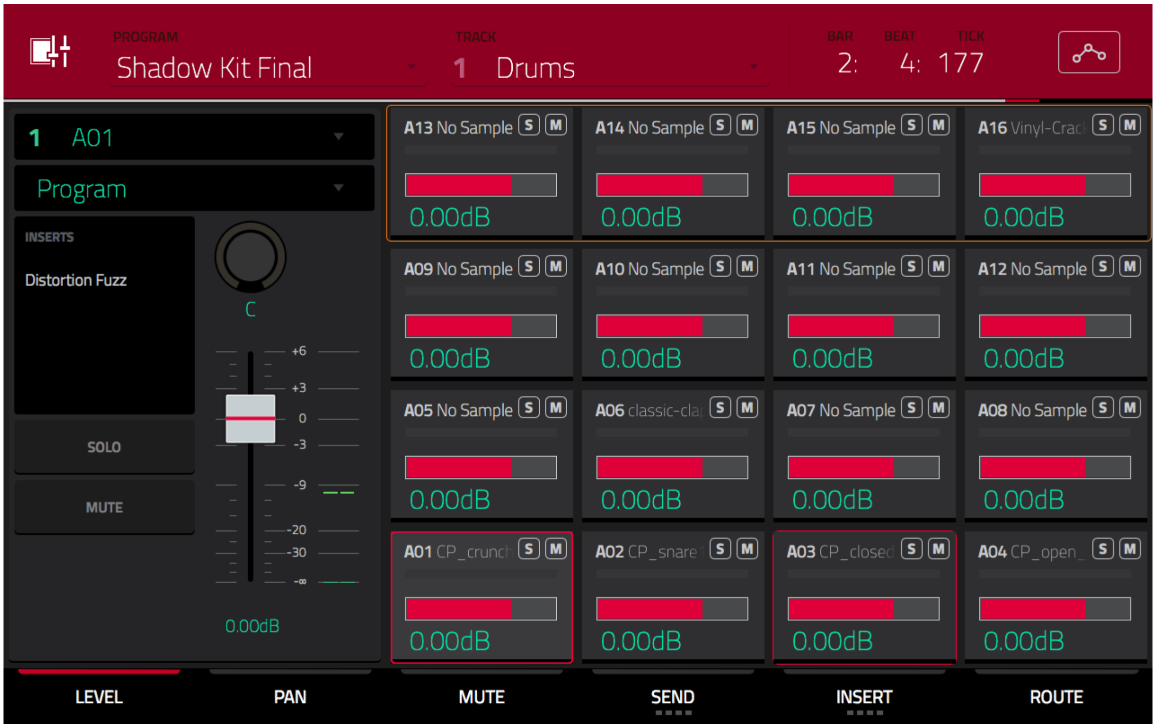
Please note that these insert effects are applied to the entire pad, so all layers are affected. But also note that this is only affecting the currently selected pad – this is not a program-wide effect (we'll look at how to apply effects across an entire program later in the book).

*To compare how your pad sounds without the effect added, tap the **ON** button at the end of the **INSERT 1** line for a temporary effect bypass.*

The Pad Mixer

An important part of building a kit is to ensure the 'mix' is perfectly set up – for example, each sound must be at the correct volume relative to the other pads. While we are able to adjust individual *layer* 'levels' in PROGRAM EDIT, we tend to adjust *pad* volumes in the dedicated **pad mixer**.

The pad mixer may look a little scary, but it's actually very intuitive and easy to use. To access the pad mixer, go to **MENU** > **PAD MIXER**:

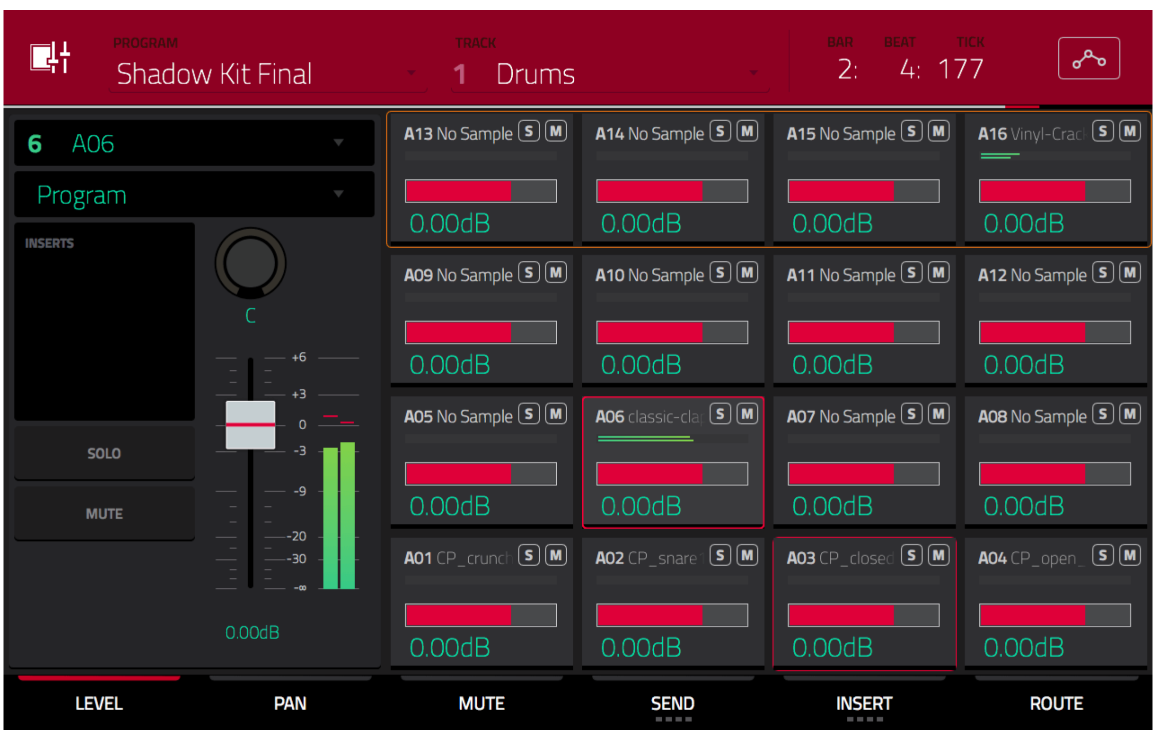


This screen effectively consists of two separate parts. On the left we have the mixer details for the currently selected pad, where we can control level, pan, solo/mute status, routing and effect inserts. This section is designed to allow you to accurately focus on the mixer settings of one specific pad.

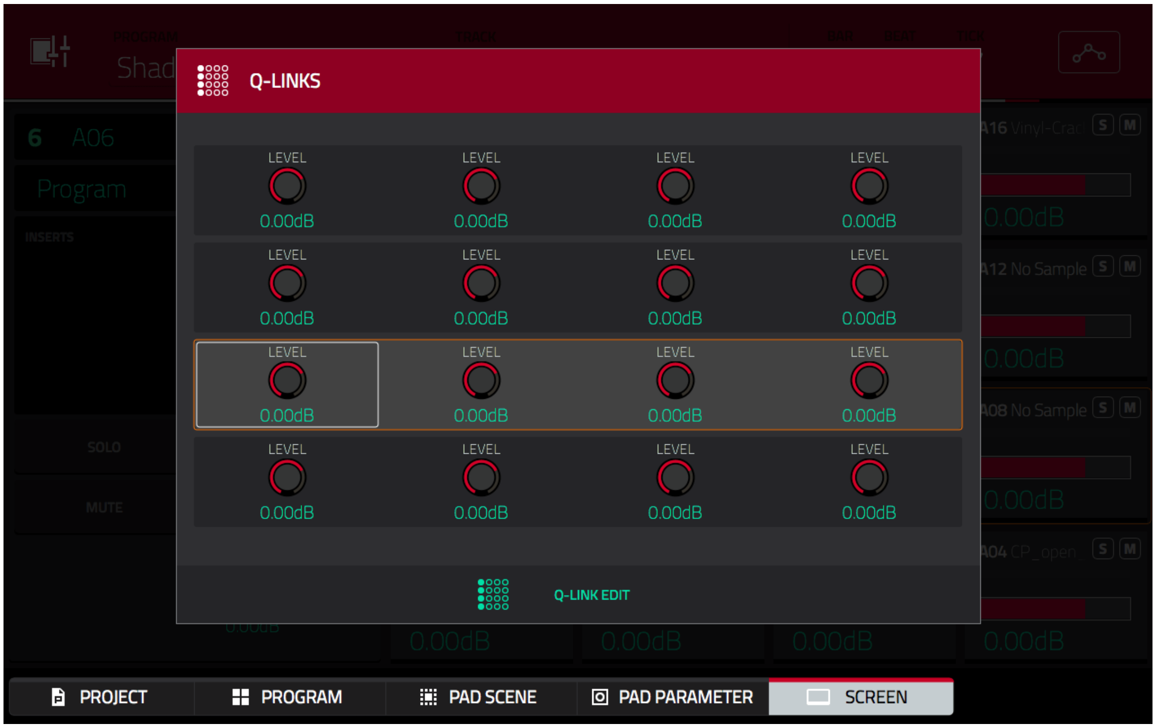
On the right we see an editable overview of all 16 pads in the current bank (with the currently selected pad highlighted). Here you can see the ‘layer 1’ sample displayed, the mute/solo status (if you mute a pad, it becomes silent, if you solo it all other pads become silent instead), level meters and an adjustable volume control. This right hand section is great for viewing an overview of your program mix.

Hit pad **A01** and you’ll see the level meter in action:

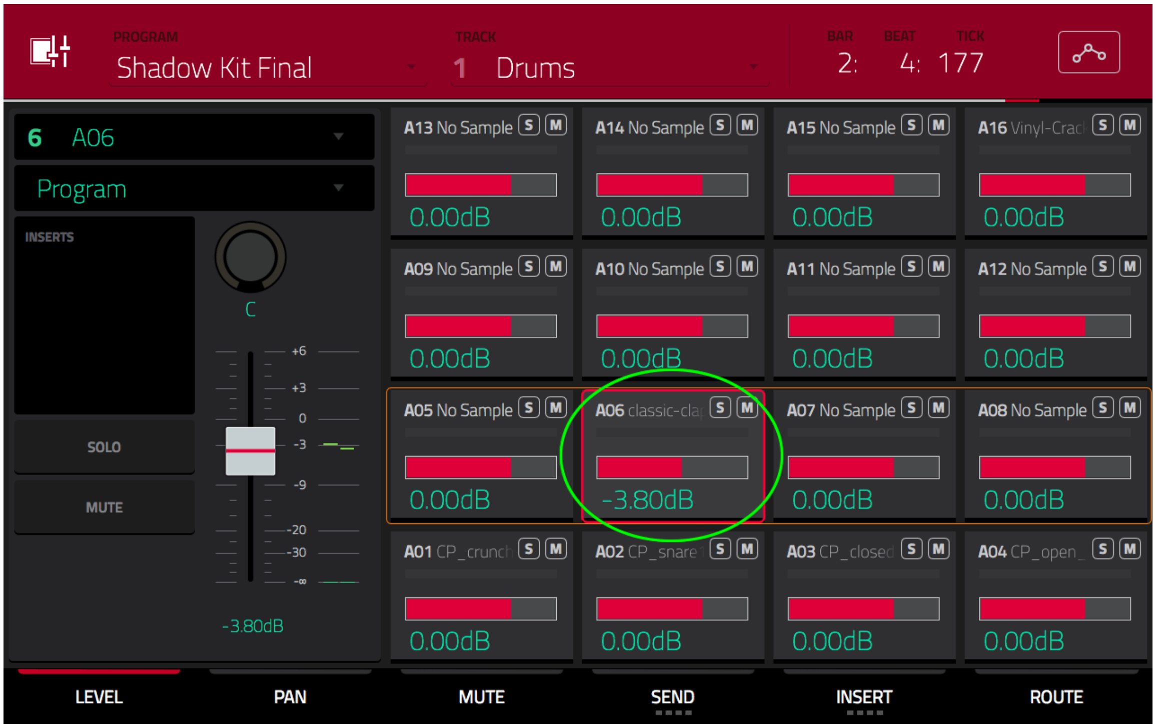
Hit all the active pads in our program. When you play the clap on pad **A06** you'll see the level meter hit red:



You should also be able to hear that this clap is much louder compared to the rest of the pads in this program. We can adjust the level of the entire pad using the big slider on the left side, or you can use the mini level meter on the right. As you'd expect, this can be adjusted with 'single tap and data wheel', a double tap to bring up an enlarged slider pop up, or a Q-Link:



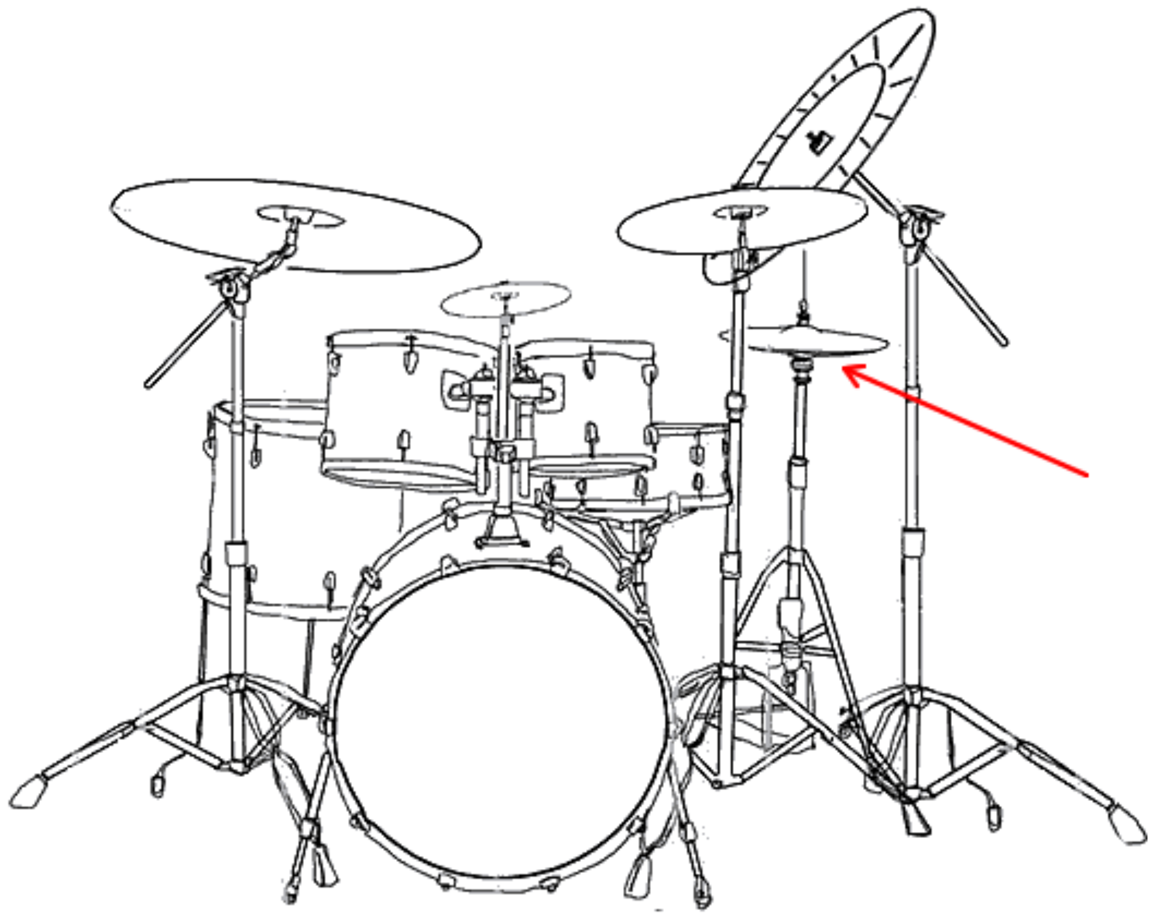
Start dropping the volume of pad **A06** (**Q6**) and while you do, keep previewing it and comparing it to the kick and the snare until you feel you have a good balance. Remember to hold down **SHIFT** for more accurate volume changes. Try a volume of **-3.80dB.**



Finger drum with your kit to hear the overall balance change – sounds good to me.

Manipulating Stereo Field

If you stand directly in front of a drum kit you would hear some instruments more loudly in one ear compared to the other. Why is this? Well, take a look at a typical kit configuration:



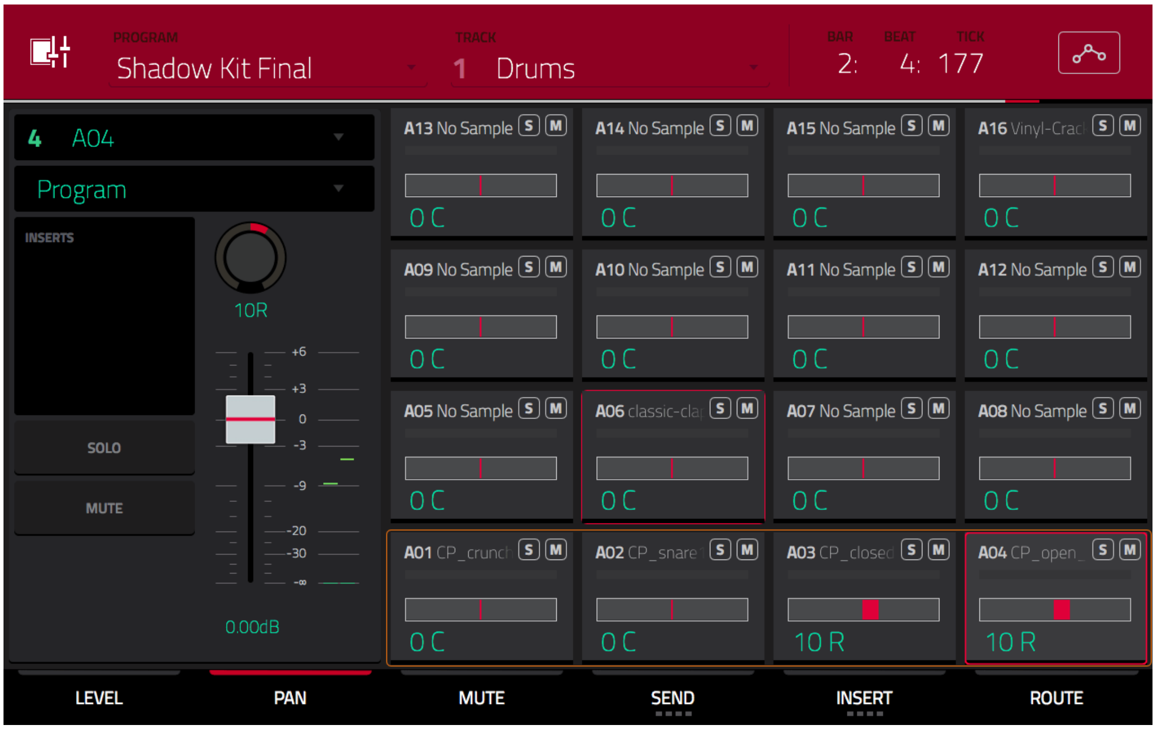
As you can see, the hi hats are positioned more to one side compared to the snare and kick, so when building a drum kit in an MPC we often tend to pan the hats slightly to one side of the stereo field.

We can easily pan our hi hats using the pad mixer. On the main **PAD MIXER** page, you have a dedicated pan control for the currently selected pad:

Alternatively, hit the **PAN** button at the bottom of the screen:

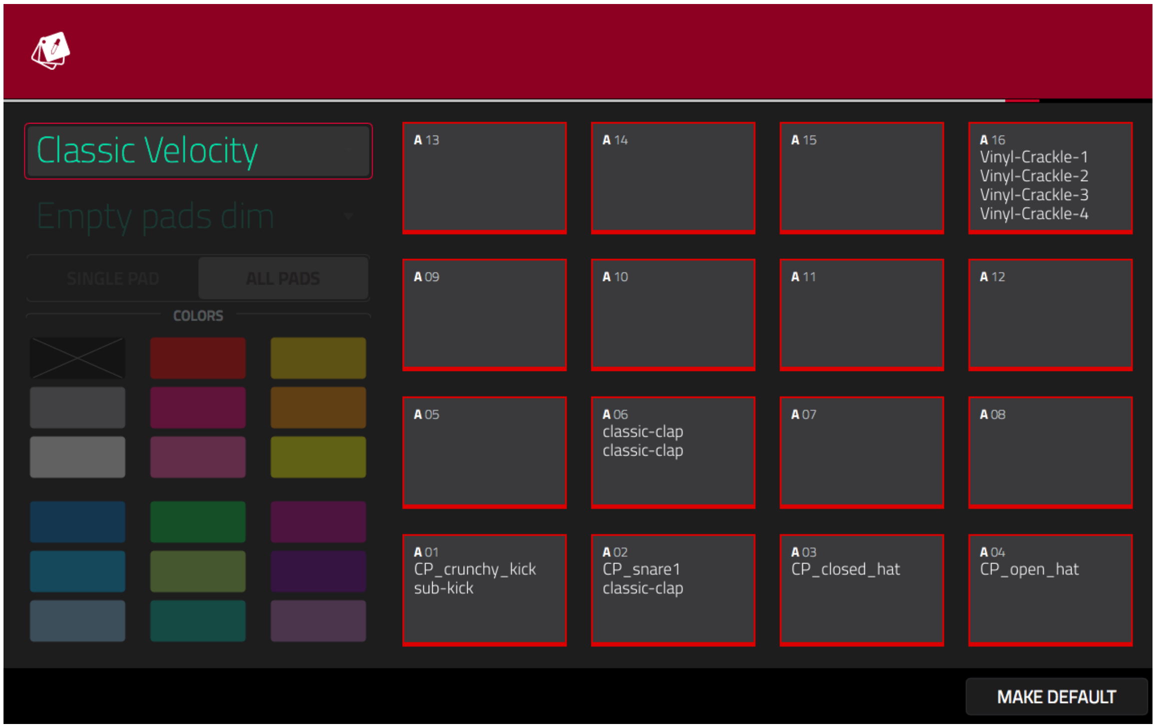
Here you can see that the right hand side of the screen now shows the panning information for all pads in the current bank. This is great if you are trying to change the pan of multiple pads.

Using either option, set the **PAN** for the closed hat on **A03** and the open hat on **A04** to **10R**:



Setting Custom Pad Colours

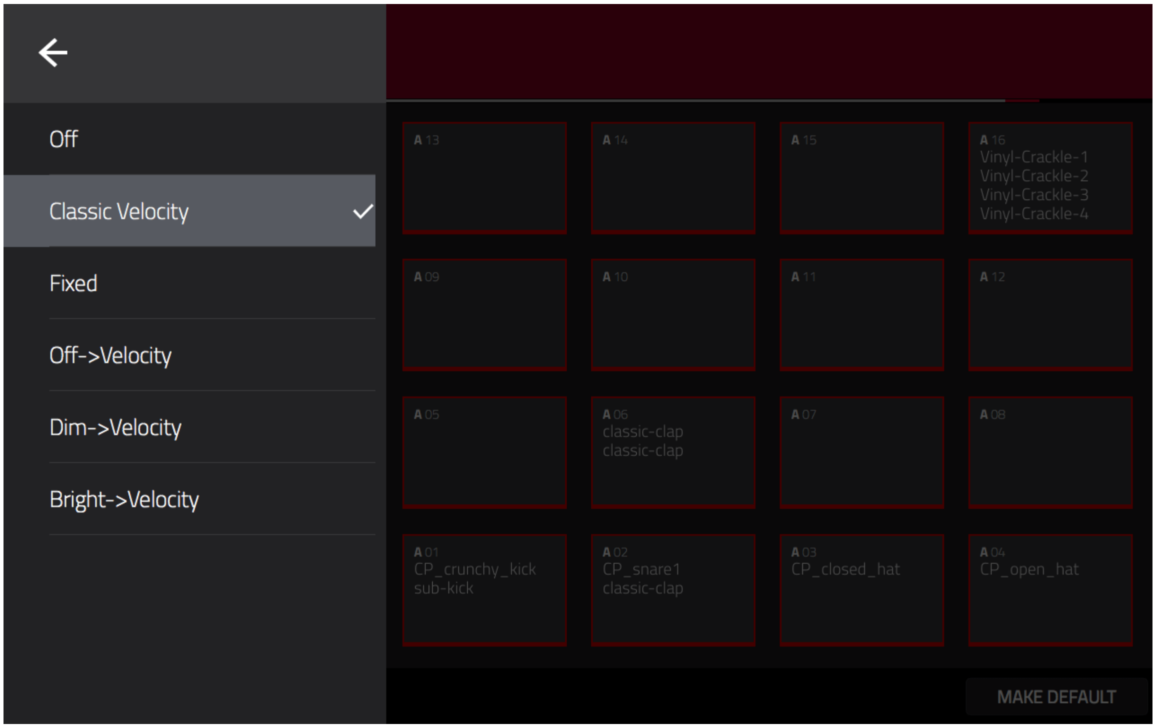
By default, your pads illuminate when they are hit, with the colour reflecting the velocity at which you hit the pads. However, the MPC offers a few ways to customise the illumination of your pads. Go to **MENU** > **PAD COLOR:**



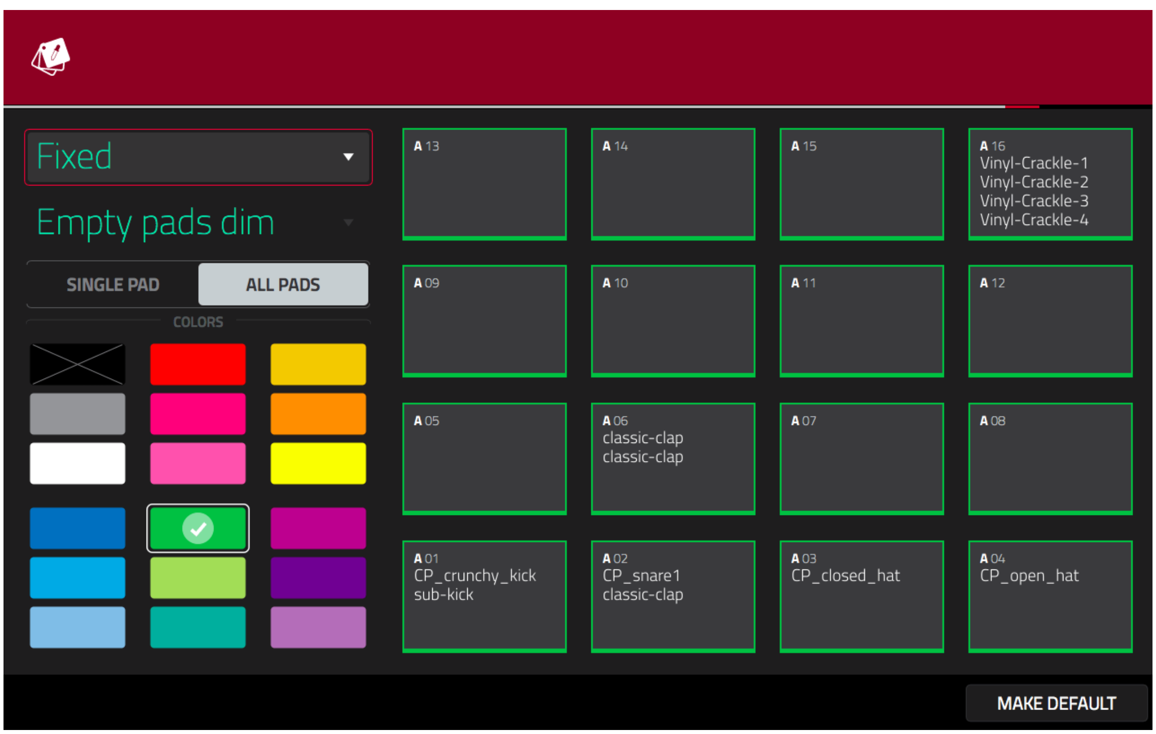
There are many combinations of options available, all of which can be applied to all the pads in a program, or to individually selected pads.

The default option is '**Classic Velocity**', in which all pads in a program illuminate when hit, the colour reflecting the velocity of the hit (yellow represents soft hits, with the colour getting progressively more red the harder you hit the pad).

Double tap '**Classic Velocity**' to see all the illumination options:



If you select ‘**Off**’ then no pad will ever illuminate, just like legacy MPCs. Now select ‘**Fixed**’:

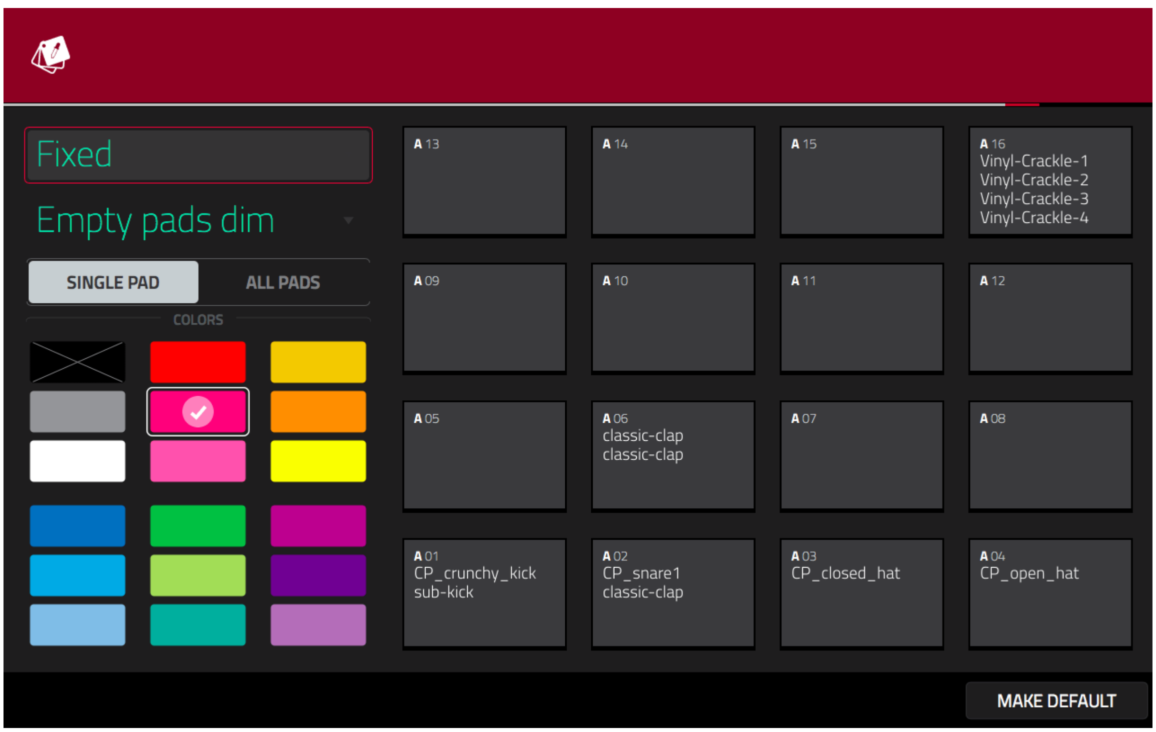


Initially, all pads will now illuminate green. If you hit the pads you'll see that the velocity you strike with makes no difference to the colour – this option is purely about setting the pads to a constant colour.

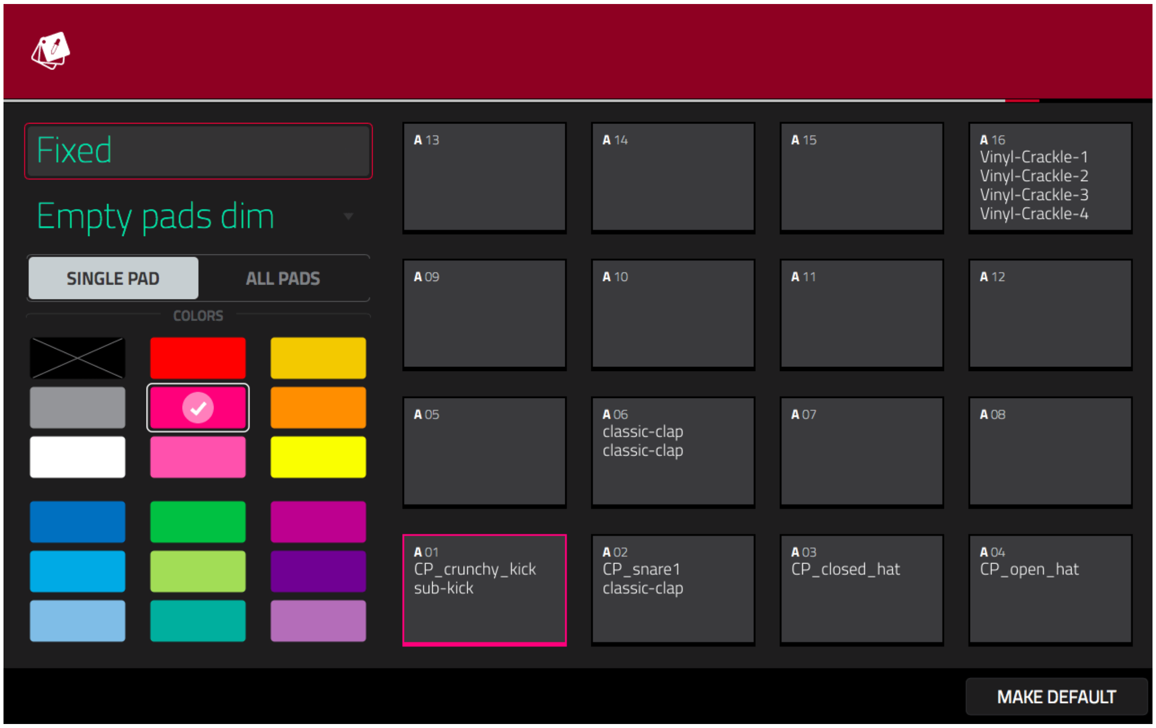
To change the colour of the pads you'll need to click on one of the colour swatches on screen – for example, click on the yellow swatch and all your pads will turn yellow.

In fixed mode you can also control how empty pads are illuminated – choose from '**Empty pads dim**', '**Empty pads off'** or '**Empty pads normal'** ('normal' will leave all pads illuminated, regardless of their status).

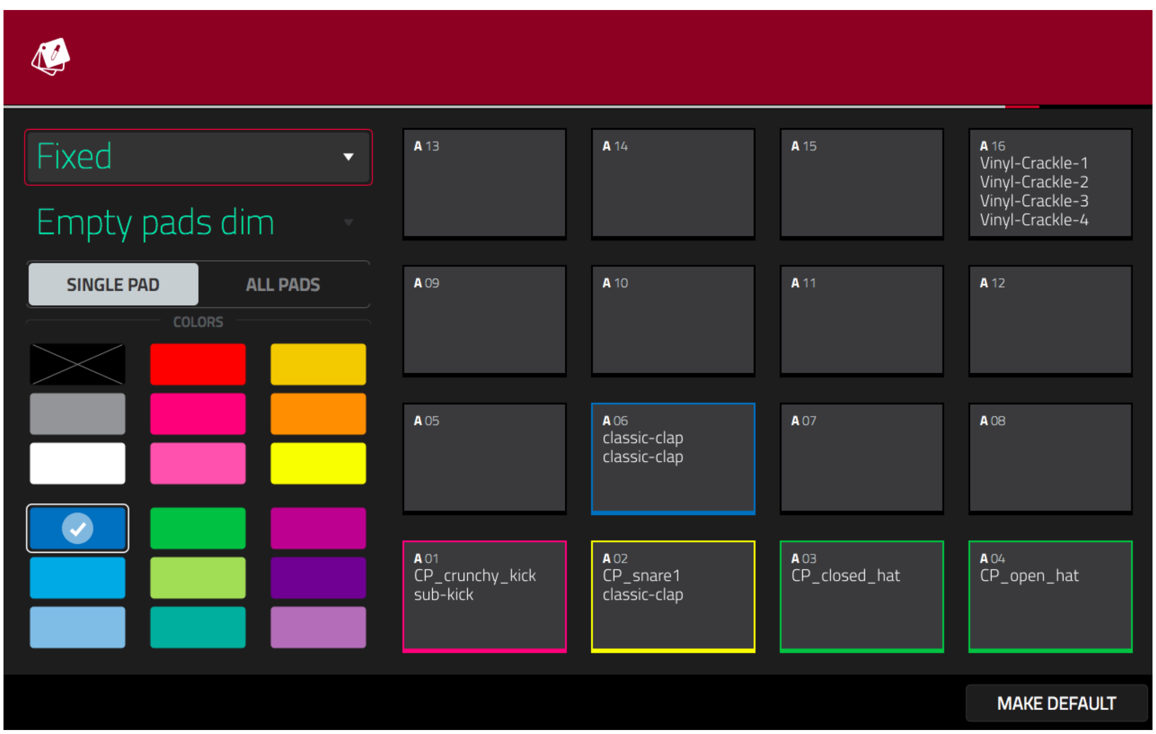
If you wish, you can assign a specific colour to a *specific* pad. This could be done to indicate the type of sound assigned to a pad. First press **SINGLE PAD**. Now select the dark pink colour swatch - you'll see that the pad assignments all go blank – this means that no pad is currently set up to be pink.



Now press our kick pad **A01**:



As you can see, these two pads are now highlighted 'pink on screen. Now select a **Yellow** swatch and press the snare **A02.** Now select a **Green** swatch and press the pads relating to the hi hats – **A03** and **A04**. Finally set the clap on **A06** as **blue**.



Your drum pads are now fully colour coded (I've left the vinyl samples un-coloured as these aren't really there to be played).

This pad lighting configuration is saved in your program's XPM file, so when you load it at a later date you'll have your lighting assignments already set up for you. You can view mine by loading the program '**Shadow Kit v2.xpm'.**

Finally, as well as 'Fixed' and 'Classic Velocity', be aware that there are even more display options available:

- '**Off > Velocity**' – all pads are initially unlit, but as you strike the pad, it illuminates in the colour you've previously set, with more brightness the harder you hit.
- '**Dim > Velocity**' – all pads are initially dimmed in the colour you've previously set, but as you strike the pad it illuminates with more brightness the harder you hit.
- '**Bright > Velocity**' – all pads are initially set quite brightly in the colour you've previously set, but as you strike the pad it illuminates with more brightness the harder you hit.

Pad colours are program specific, however if you wish you can set your favourite pad colour layout to be the 'default' colour layout for all programs. Just tap **MAKE DEFAULT > DO IT**.

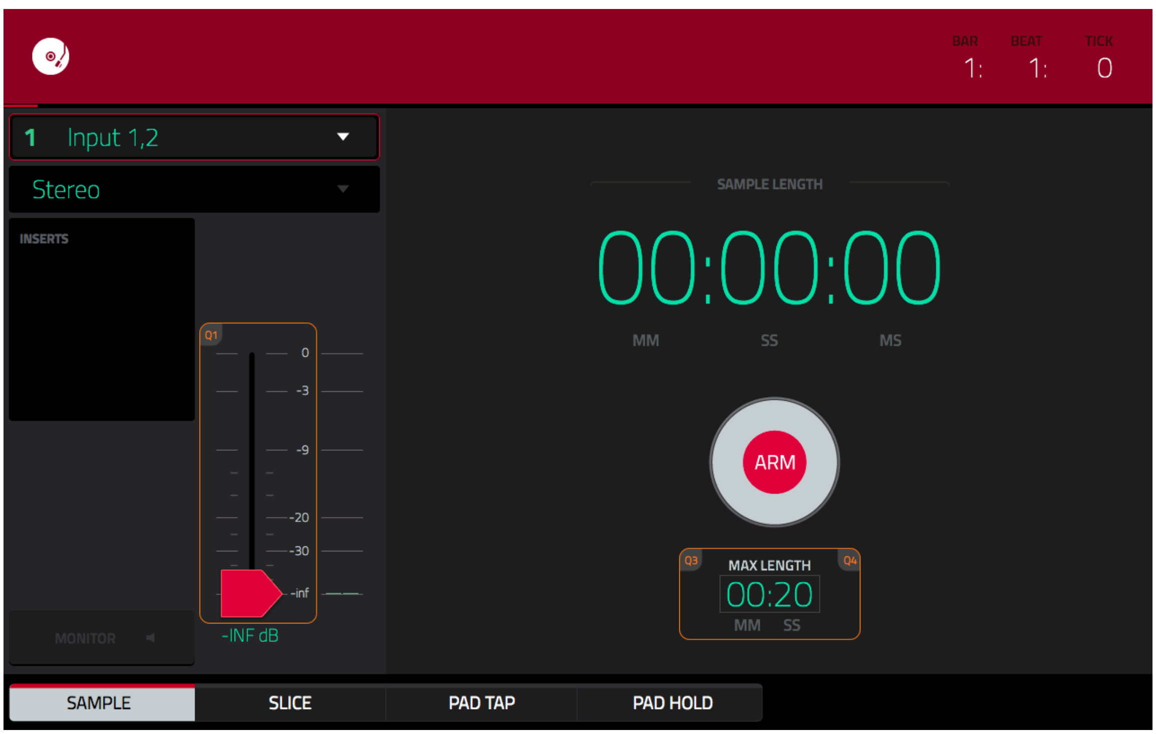
008 Sampling 101

The MPC allows you to record audio from any source, be it vinyl, CD, phone, computer, synth or microphone based recordings of drums, vocals, guitar, piano etc. In order to record any audio source into your MPC you'll need to connect the sound source to the recording inputs of your audio interface.

*Please refer to **Appendix A** at the back of the book for information on how to connect the various different types of sources.*

Sampling Audio into the MPC

Later in the book we'll look at how audio can be recorded directly to a sequence's dedicated audio track, but let's first look at the more traditional MPC method of 'sampling'. Sampling is performed in SAMPLER mode, so hit the **MENU** button and tap on the **SAMPLER** icon (if you have an MPC X, hit the dedicated **SAMPLER** button).

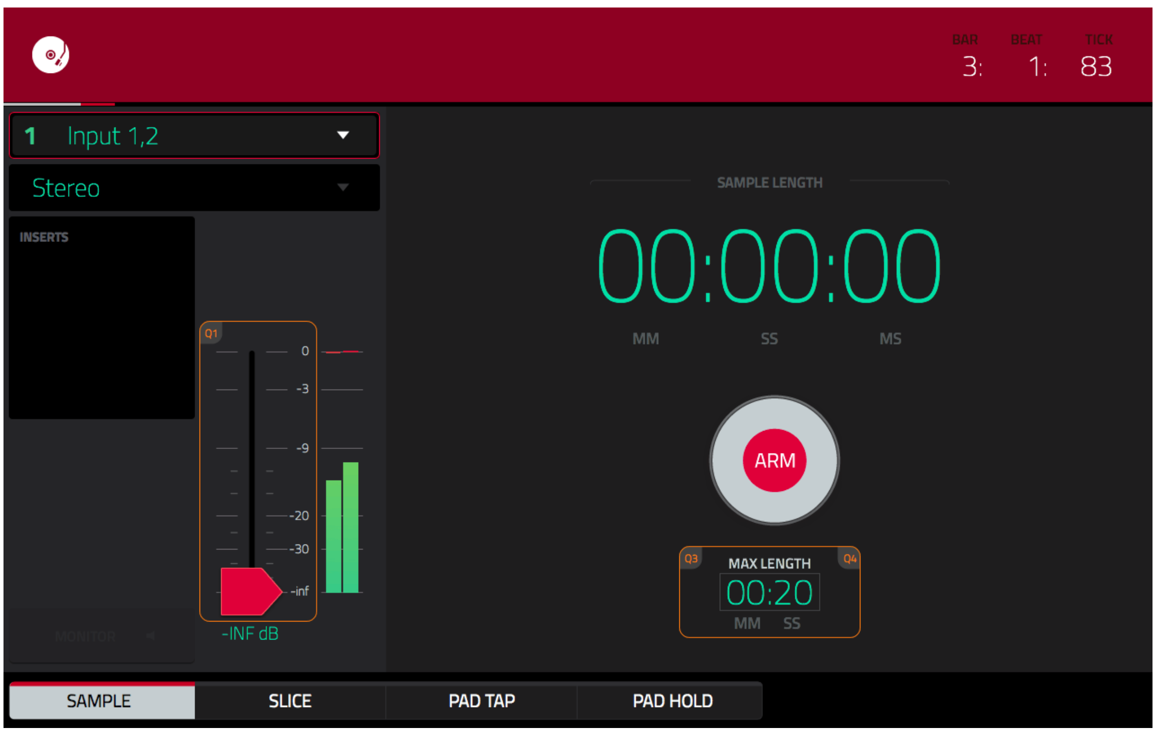


Let's look at some of the parameters we have here. Our '**input source'** is set to '**INPUT 1,2**' which means that the sampling screen will be monitoring the incoming audio from inputs 1 and 2 from your audio interface (The Live hardware label refers to these as L & R). You can double tap this to select alternate inputs for recording if you need to; you can also select a single input rather than a stereo pair.

Underneath is '**Output**, which defines whether the recorded sample is either '**Mono'** or '**Stereo'**, and in normal usage should match the number of channels used in the 'Input' section. So, if you were sampling a mono (single channel) source such as a microphone or electric guitar, you would connect the microphone to the '**L**' recording input, select '**Input 1**' as your 'input source', and '**Mono'** as your 'output' (there is normally no benefit in creating a stereo file from a mono source).

If your sound source is truly stereo, that is, a signal that contains panning information, such as a complete song or most drum loops/breaks etc, then you need to connect the stereo source to both ***L*** and ***R*** recording inputs, so set '***Input'*** to '***Input 1,2'*** and set the 'output' to ***'STEREO'.***

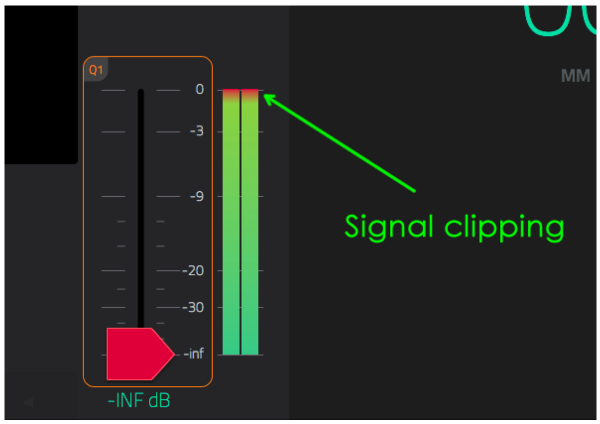
Level Meter



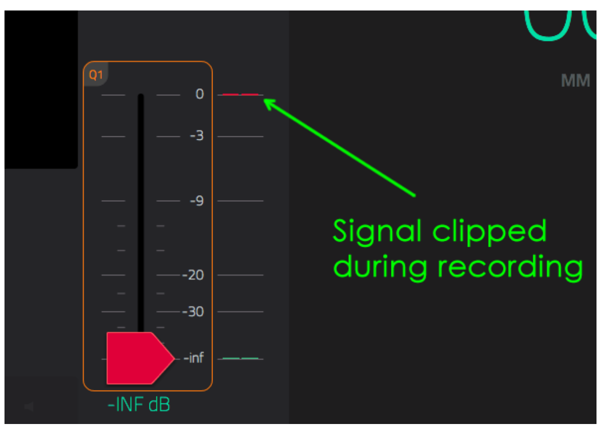
The '**Level Meter'** shows you the level of the incoming sound source so you can control it to avoid distortion during sampling. To control the level of the incoming audio signal, you can either use the volume control on the audio source (if it has one) or use the input gain level control on your audio interface – on the MPC Live this is the **REC VOL** knob at the back of the unit, on the MPC X this is controlled via the GAIN knob(s) on the top of the unit.

Firstly, turn the **REC VOL** knob to a fairly low setting and play your audio source into the MPC – if you are playing an instrument or recording a vocal, try to play/sing at the loudest volume you'd expect to play at during the performance. The meters on the page should start falling and rising with the music.

Turn the **REC VOL** dial up until the music starts distorting, and then edge it down until the distortion stops. This is your maximum recording level. At the maximum level, the sample meter should be fluttering around the end of the scale, and not solidly sticking to the end.



You may notice that after an audio signal has passed through your sampler, the peak level graphic will now contain additional coloured lines:



The red lines at the top indicate the highest level your signal ever got to, and is useful for judging whether or not your signal is reaching the 'hot' signal point – in the example above, it is clear the signal had clipped. The next time you sample, these peaks will remain there unless you first return these lines to the default position - to do so, just double tap the meter.

What About Signal-To-Noise Ratio?

You may have heard a term called 'signal-to-noise ratio'. This simply refers to the level of background noise compared to the level of the actual (direct) signal you are trying to sample. Basically no matter what analogue source you record from, you will experience an amount of background hiss that stems from the components used in the sound source, as well as from the audio cables you use (and also from the actual MPC interface).

As you increase the REC VOL on your MPC, you do not only increase the level of the sound you want to sample (the 'direct' sound), you also increase its background noise level. Hence it is important that you achieve the loudest sound possible from the incoming sound source - typically by increasing the output volume of the sound source (if possible).

Monitor

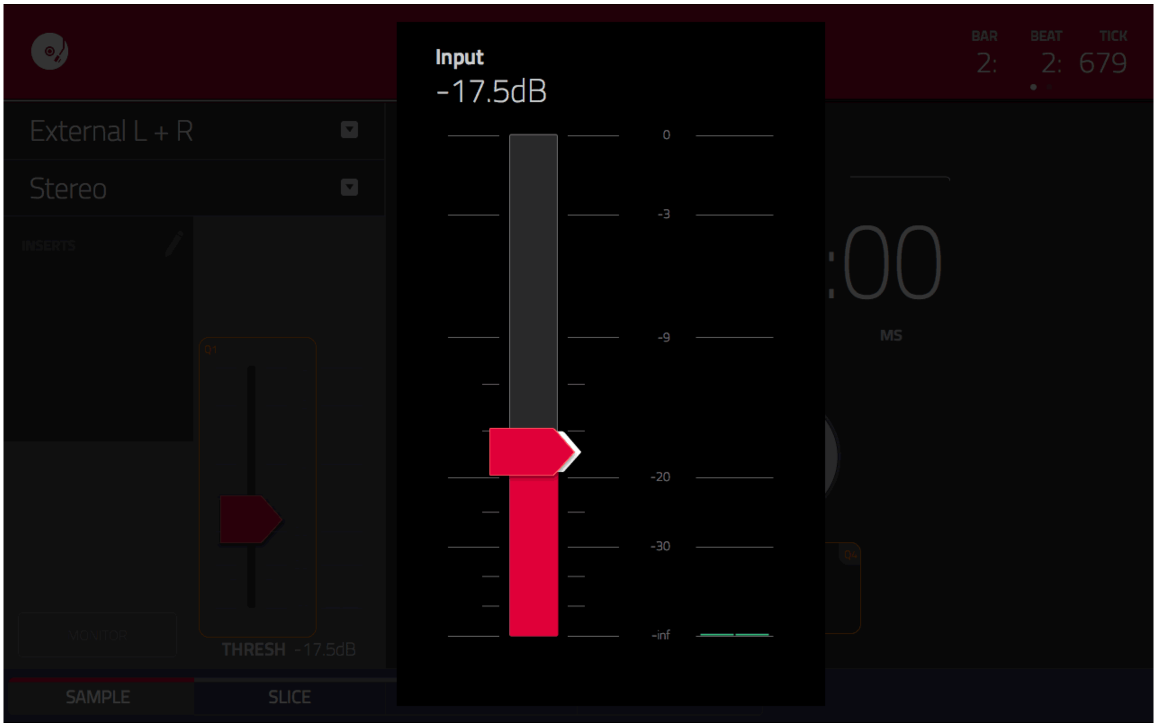
When sampling live performances, you'll often want to monitor the incoming audio using headphones to hear how the performance sounds on top of any background track it's being recorded on top of (e.g. click track etc).

With **MONITOR** turned **on** ('white'), you'll hear the incoming audio *before* it is processed by the MPC – this will avoid any processing latency and will give you the most accurate impression of the timing of the performance compared to the click track.

Threshold

The **Threshold** level helps us automate sampling a little as it lets us initially 'arm' the sampler, making it instantly ready to record audio the moment that the incoming signal's loudness exceeds the threshold level you set. Most of the time set it a little above the background noise level, that is the general 'hiss' that is present when recording from analogue sources, or in the case of vinyl, the static and crackles.

To set a threshold level, drag the red threshold pointer up or down with your finger. If that's too fiddly for you, double tap the slider to bring up an enlarged version:



If you prefer, you can just touch the threshold slider (to select it) and turn the data wheel – for more accuracy, hold down the **SHIFT** button on the MPC hardware and the slider will move in much smaller increments. Alternatively, instead of having to first 'touch' the slider, you can just turn **Q1**.

Sampling Time

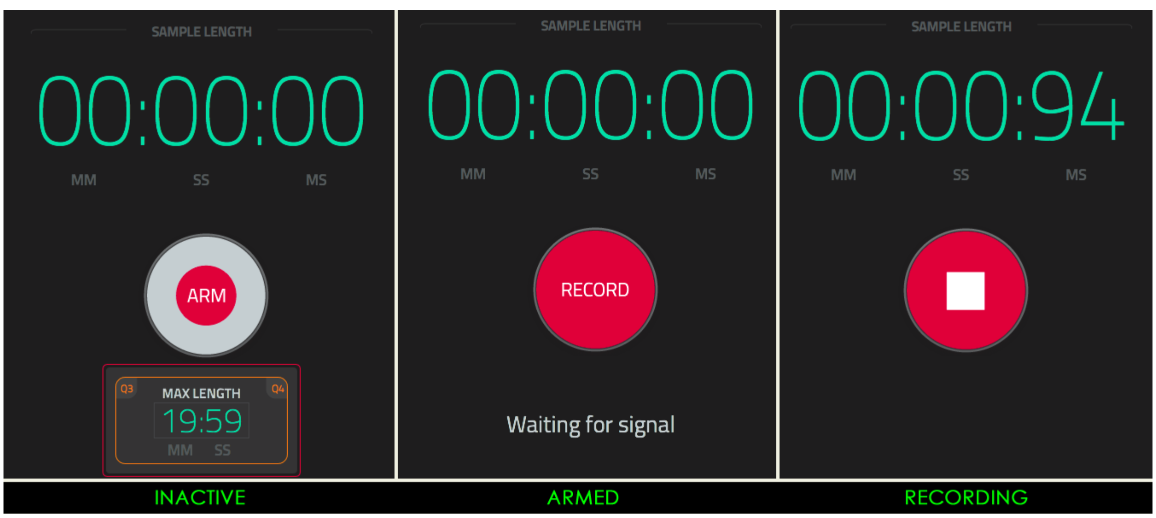
The **'Time'** field, which we've just touched upon in the previous section shows you the maximum sample time you are dedicating to this particular sampling session. Sampling time is very useful if we are sampling notes from an instrument and we want them all to be say, 5 seconds long. Simply set the time to 5 seconds and the MPC will only sample each note for that specified time.

If you are sampling long passages, or maybe performing something directly into the MPC (like a vocal), I find it best to set the sample time to a nice long setting (e.g. several minutes), or even the maximum (19 minutes, 59 seconds) – that way you can be sure that the sampler isn't going to stop recording while you are still performing.

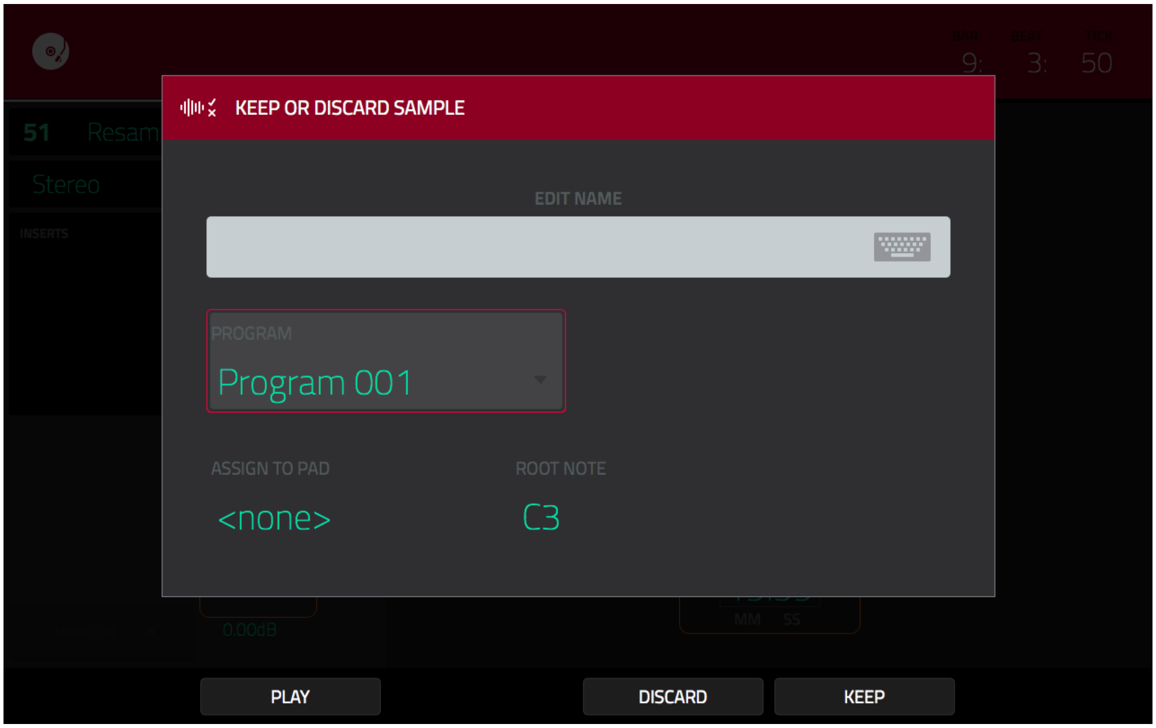
The '**Sampling Time'** is controlled by two other Q-Links; **'minutes'** is controlled by **Q3** and '**seconds'** by **Q4**:

Sampling your Sound

Once you have configured all the settings you can record your sound source. Press **ARM** to put the sampler into 'ready' mode (the wording changes from ARM to RECORD) – the MPC sampler will now wait for an incoming signal that exceeds the threshold level you previously set – upon detecting that signal, it will immediately start recording. Alternatively, if the sampler is already armed, you can press **RECORD** to manually trigger recording. Press **CANCEL** at any point to disregard the entire recording procedure.



If you have set a fixed sampling time, you can let the MPC stop the sampling automatically. Otherwise to manually stop the sampling at any point you desire, press the square '**STOP'** button After a successful sampling session, you will see the following screen:



Press **PLAY** to preview the recording. If there's an issue simply select **DISCARD** and you can start the sampling process again. Otherwise, press **KEEP**.

Now, it's always a good idea to name your samples rather than using the default '**New Sample**' name. Touch the 'keyboard' icon to the right of the current sample name to bring up the keyboard, enter the new name and hit **DO IT**.

The '**ASSIGN TO PAD**' option allows you to assign this sample to a pad in the current program; let's just leave this to **<none>**.

The '**ROOT NOTE**' option tells the MPC the musical pitch of the sample (if it has one) – I'll cover this later in the book. Just leave this set to the default **C3**.

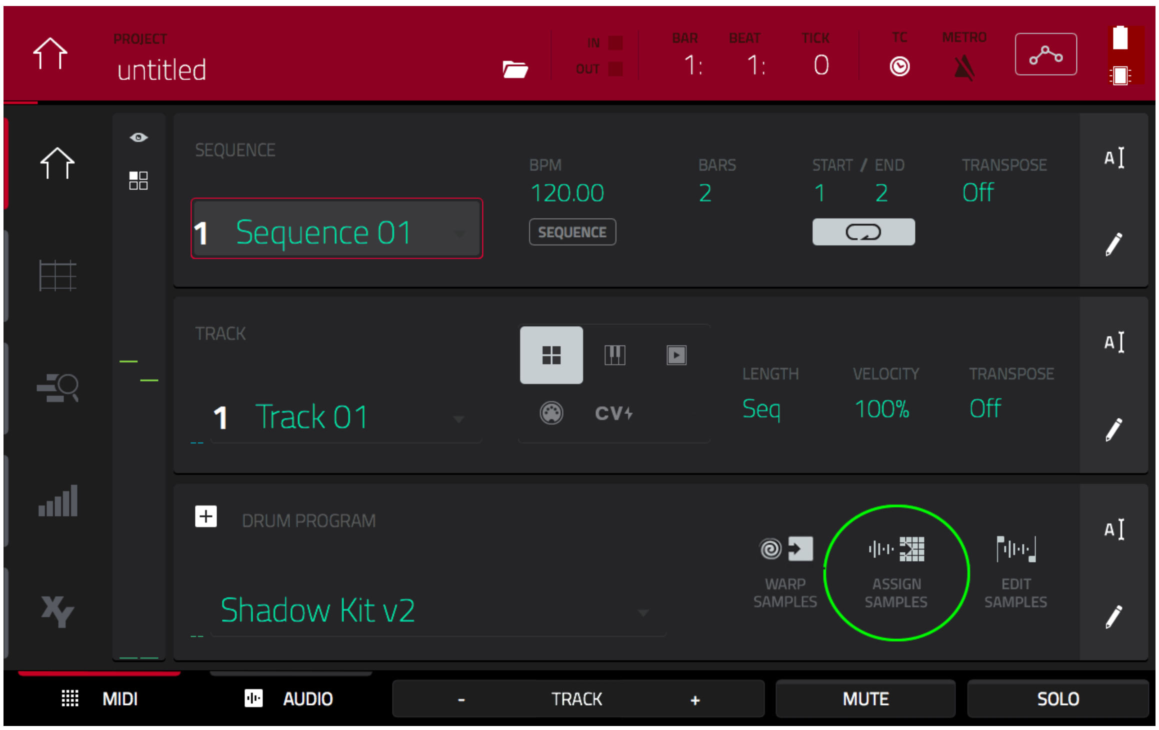
Once you've finished naming your sample, press **KEEP** and your new sound will be automatically placed in your project, ready for use via the Project Information panel. With your new sample now converted to a digital file you will normally need to perform a little bit of 'housekeeping' to remove dead space from the beginning and end of the file. We perform this work in SAMPLE EDIT mode.

Sample Edit Mode

Sample Edit mode is where you get a chance to perform a number of different processing functions to any of the sounds loaded into your project, be it functional tasks like 'tightening' up the start and end of the sample, right through to more creative stuff like chopping and time stretching.

Make sure the **Shadow Kit v2** program from the previous chapter is currently loaded into your Project and selected in **MAIN** – if it isn't, you can load it from the **008** folder.

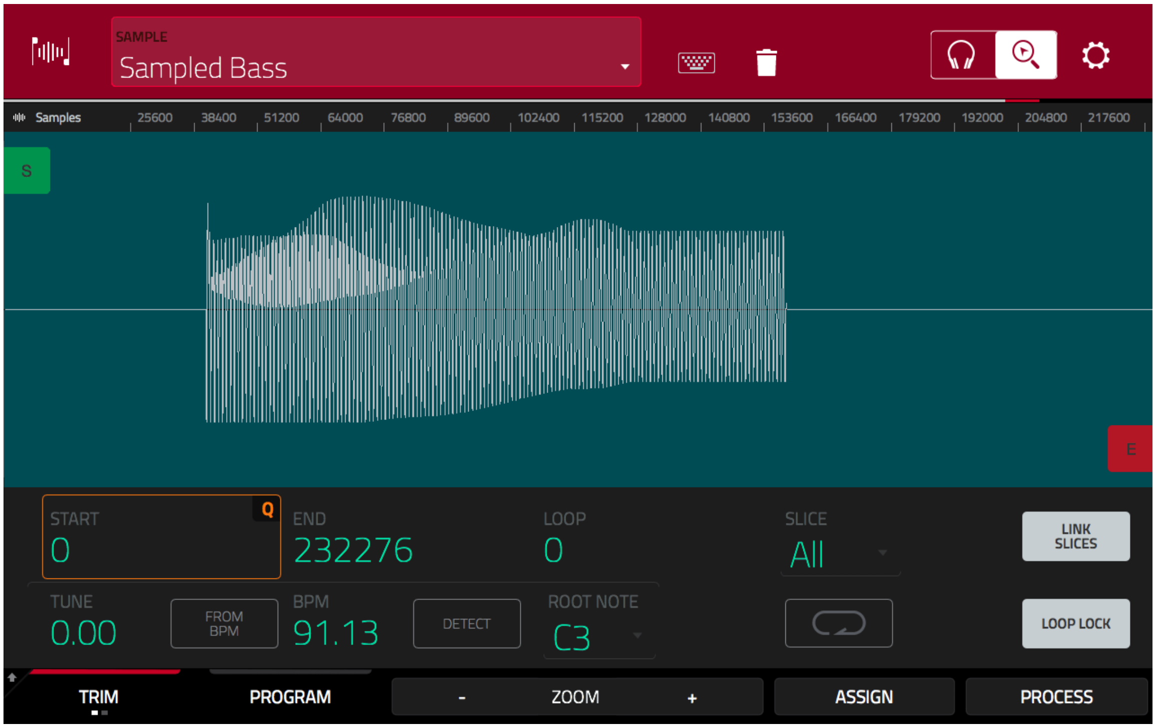
From **MAIN**, hit pad **A05** of the **Shadow Kit v2** kit and press **ASSIGN SAMPLES**:



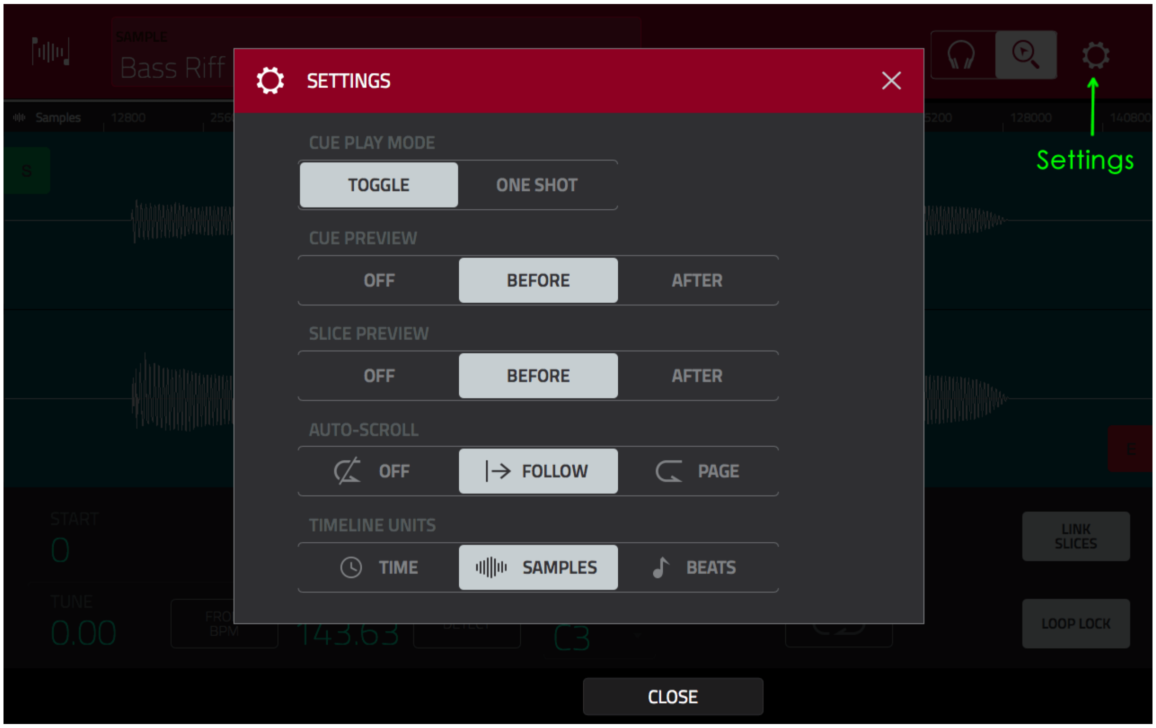
Hit **BROWSE** and from the **008** folder, load the '**Sampled Bass'** sample (it will be simultaneously assigned to pad A05). Go to **MENU > SAMPLE EDIT** (or use the dedicated **SAMPLE EDIT** button in the MPCX).

Waveform Editing

In SAMPLE EDIT mode you'll see a graphical representation of the sample's waveform on your screen.



The grey bar just above the waveform displays the length of the sample in one of three optional units; '**Samples'** (the standard way of measuring the length of audio samples), '**Time'** (measured in milliseconds), or '**Beats**' (this uses sequencer time of 'beats:bars:ticks' and is dependent on the BPM set in your current sequence. To select the preferred time units press the **Settings** icon located at the very top right of the screen (the gear icon):



Normally we would tend to use ‘**SAMPLES**’. Press **CLOSE** to return to the main SAMPLE EDIT screen.

The actual audio is represented by the light lines and filled areas on screen. The taller the waveform peak, the louder these sections of the audio file will be. Areas with no obvious waveform are typically silence or extremely quiet sections of the audio file. As you can see from the waveform, our ‘Sampled Bass’ sound has a portion of silence at the start and end of the sample, which is quite normal when dealing with audio recorded from an external source.

To preview your entire sample from the very beginning to the very end, press **pad 9** on your MPC.

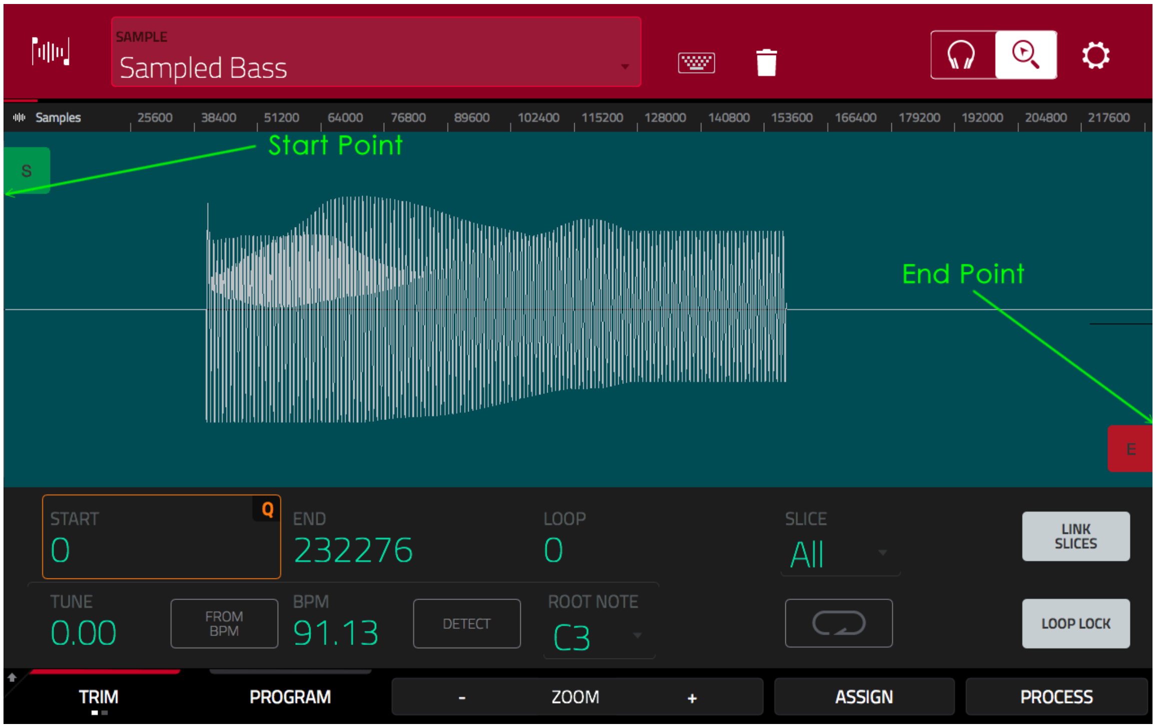
Edit Points

The play back of your sample is defined by three unique ‘**edit points**’.

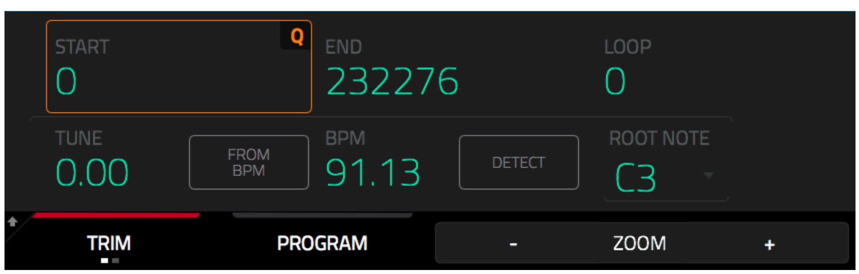
- **Start Point (START)** – this tells the MPC where to begin playback of audio.
- **End Point (END)** – this tells your MPC where playback must stop
- **Loop Point (LOOP)** – this allows you to set a point where the sample should start looping playback.

For the moment, ignore the Loop point, it’s a bit more of an advanced feature so we’ll look at this later in the book.

When you load a standard audio file (one that has never been edited by an MPC), the default '**Start'** point is at the absolute beginning of the file at 0 (zero) samples, and is shown by the green '**S**' flag marker, while the **end point** is set at the very end of the file and is represented by a red '**E'** marker:



To find out the current value of any edit point, you can look at the bottom panel:



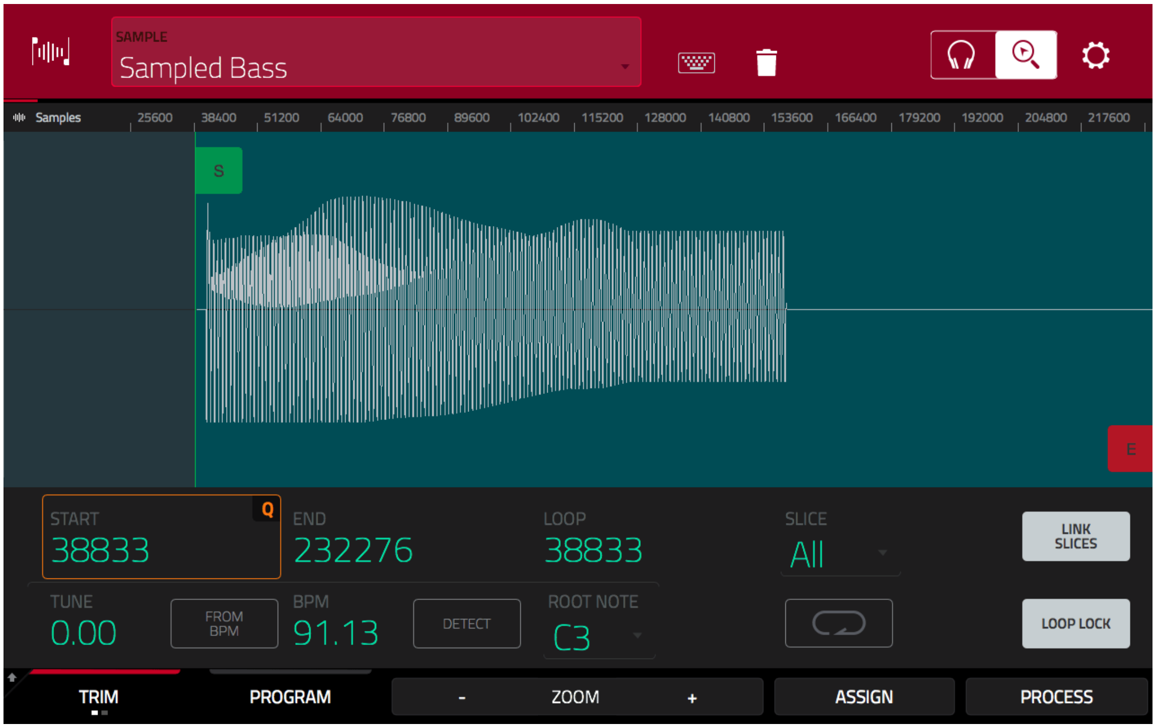
There's a few different settings here, but at the stage we only need to look at 'START' and 'END'. As you can see, the '**START'** point is **0**, and the '**END'** point is **232276**.

It should be clear from looking at the waveform that the actual audio itself in this sample does not currently start play back at the current sample 'start' point (0). As I showed you previously, a straight horizontal line with no obvious peaks is either complete silence or represents a very, very quiet section. So currently the sample is initially playing back with a brief moment of silence. This kind of gap is very common when you have manually recorded a sample in the SAMPLER screen.

This isn't ideal because when we come to trigger the sample later we would want the audio to play back *instantly*. To fix this issue, we have to adjust the 'Start' point of our sample so it lines up at the point where the actual audio begins – i.e. immediately in front of where the first waveform peak is:

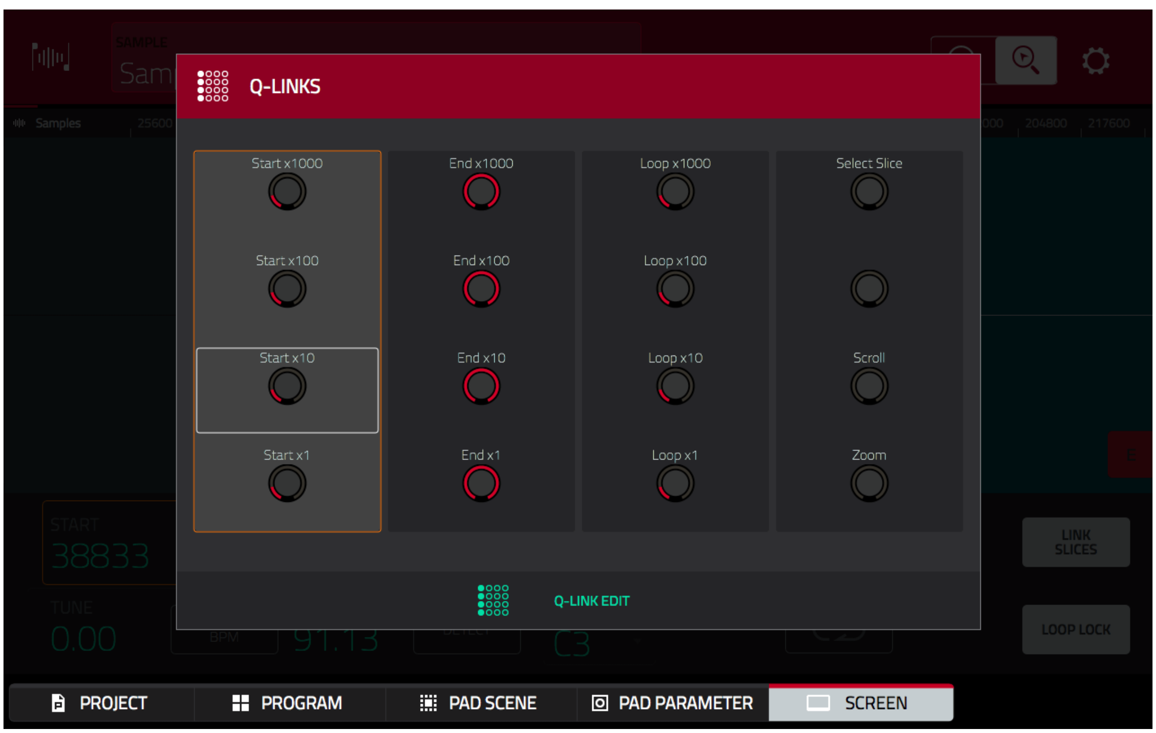
Adjusting the Start Point

There are a few ways to adjust the START point. First you can place your finger on the green 'S' flag and drag the start point marker across the screen to your desired position in the waveform (notice how the green flag turns light grey when dragging):

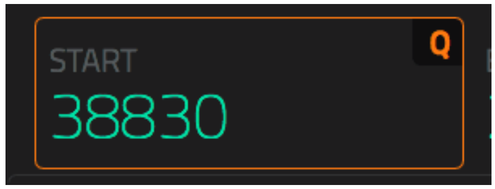


Alternatively you can first single tap the numerical START point parameter itself to select it and then turn the data wheel, but we'll look at the data wheel option in more detail when we look at the zoom function a little later.

You can also use the Q-Link dials to make start point adjustments.

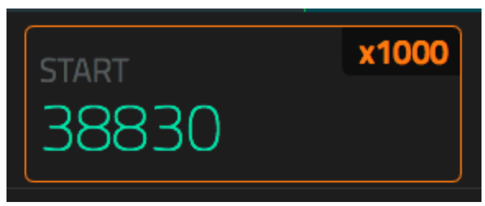


Q-LINK column 1 controls the START point in varying degrees of resolution. Notice how the 'START' point parameter box is enclosed by an orange rectangle with the 'Q' icon?



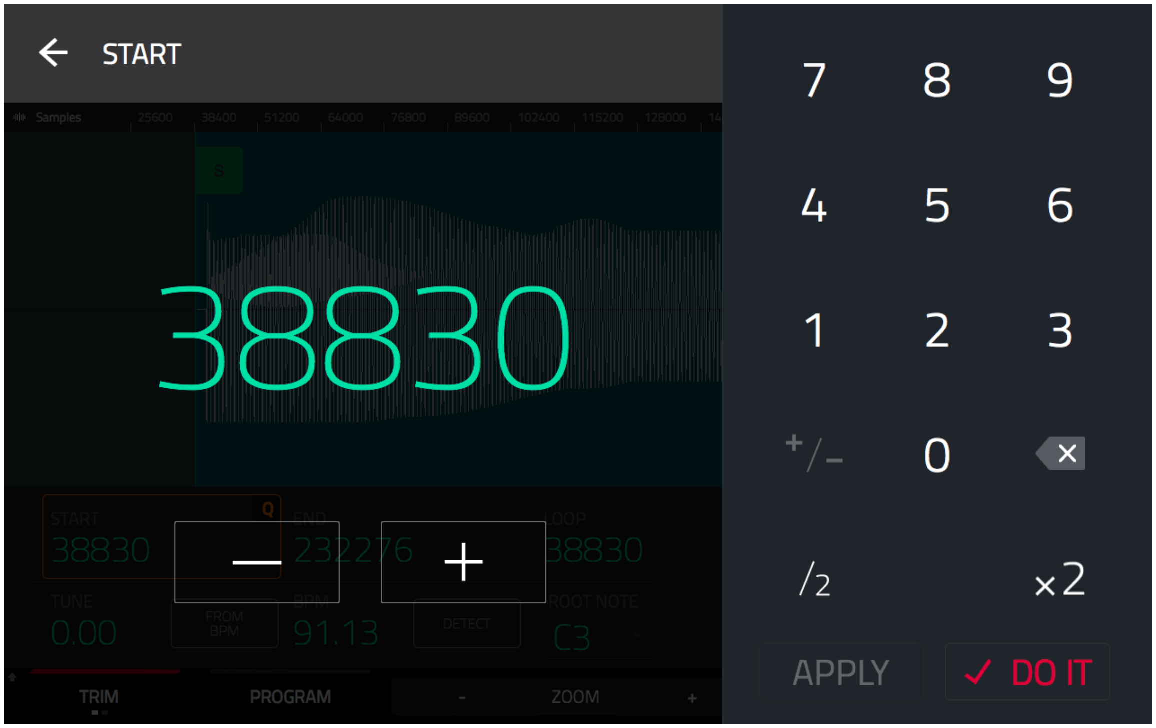
As I mentioned in the previous chapter, this orange box and 'Q' icon indicates a parameter that is editable by the Q-Link dials. In this instance there's no indication which Q-Link dial is to be used, and that's because you can use *all four* Q-link dials to edit the start point.

Lightly touch your finger on top of the **top Q-Link** dial (**Q13**) and notice how the 'Q' graphic changes:



This tells us that turning this dial will move your start point in units of 1000 samples at a time. Do the same for the other three Q-Links in that column and you'll see that the Q9 moves the start point 100 units at a time, Q5 moves 10 units and the bottom Q1 moves 1 unit at a time. So for moving your start point over large distances, use the top two Q-links, then use the bottom two for finer editing.

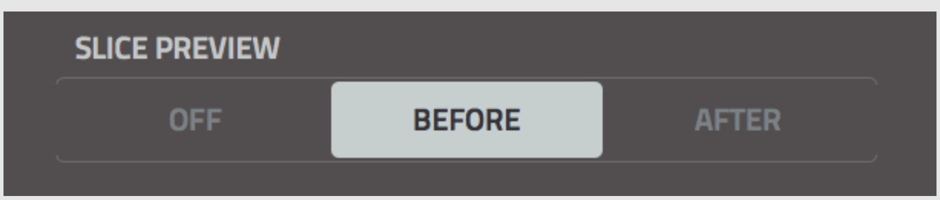
Also note that you can double tap the start point to bring up a number pad to enter a specific numerical start point:



Whichever method you prefer, set your start point roughly where the waveform peak begins – approximately **38830**.

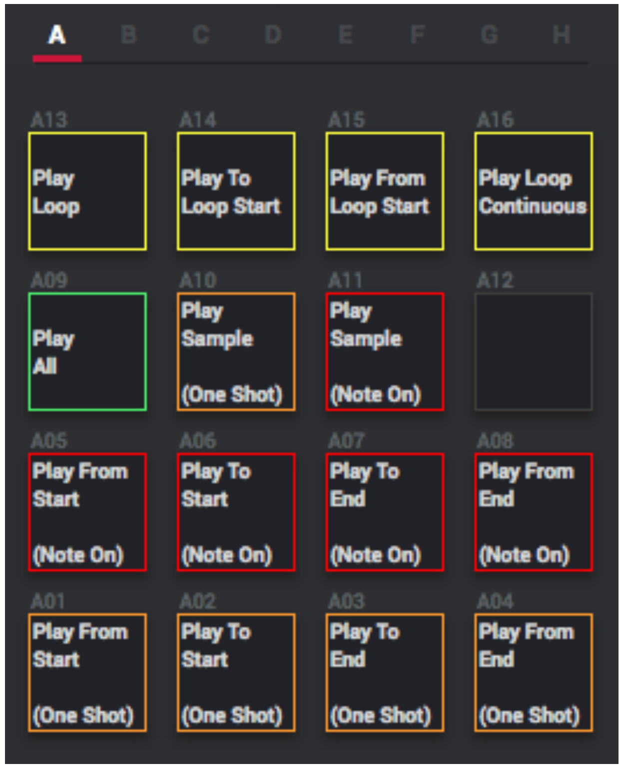
Scrubbing Audio

If you like to be able to hear the audio while you adjust an edit point, hit the 'Settings' icon (top right of screen) and set '**SLICE PREVIEW**' to '**BEFORE**' and hit **CLOSE**.



Now when you adjust any edit point you'll hear an audio 'scrub' which plays back the audio immediately before the current edit point.

Want to hear what you current edit sounds like? In SAMPLE EDIT each of your MPC's rubber pads is dedicated to a specific type of audition playback:



If you wish to hear the audio *between* the start and end points currently set, press your MPC's **pad 10** (**Play Sample, One Shot**). So press pad **10** and you'll observe that playback begins from the new *start point* of **38830**, not 0. **Pad 11** also previews the same audio region as pad 10, however this pad is set to 'Note On', which means it only plays back for as long as you hold down the pad – simply pick the type of playback you prefer.

Pad 1 is the '**Play From Start'** pad (One Shot version) and this will just play 2 seconds of audio from the current start point.

However, press **pad 2** and you hear nothing – this is the '**Play *To* Start'** pad and will only play the 2 seconds of audio *up to* the current start point, which in our case is silence.

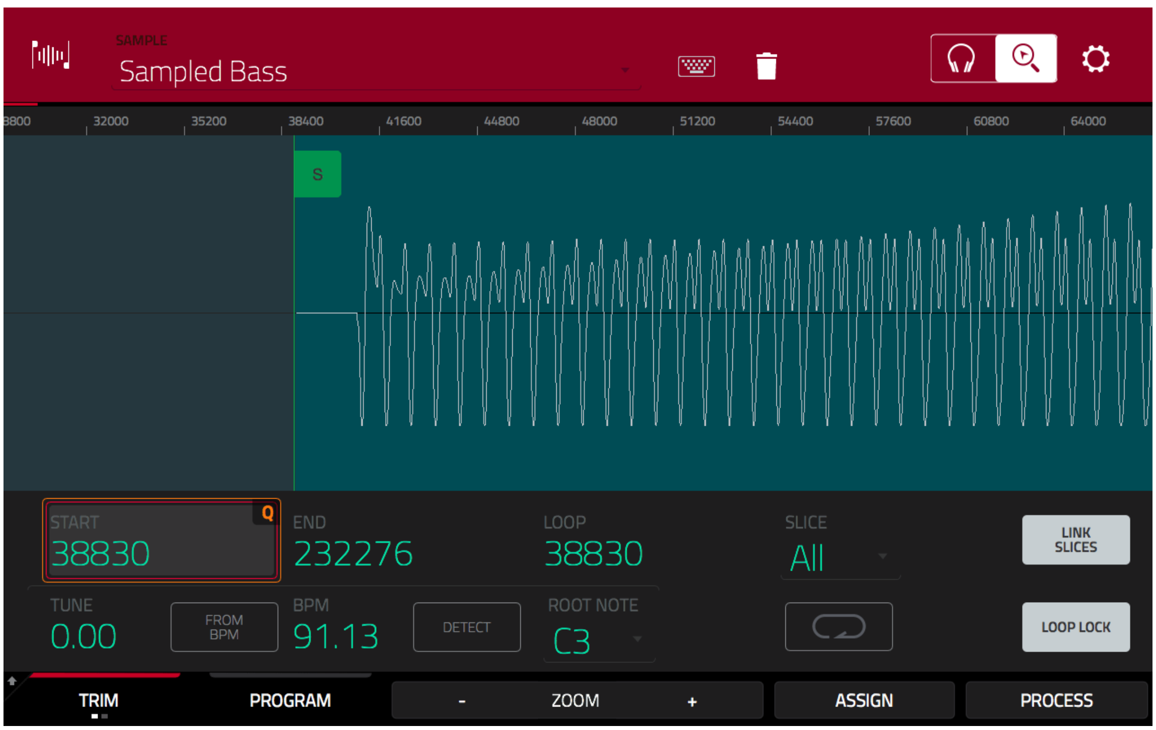
Now try previewing with **pad 9** (**Play All**). You should hear that this 'Play All' preview *ignores* your newly set 'Start' point and just plays your entire sample from '0' all the way to the end.

Zooming In For Accuracy

Now, while everything already sounds pretty good, we can optionally get things much more 'technically' accurate. To do this, we need to magnify the waveform display by 'zooming in'. One way to do this is to use the familiar 'pinch and zoom' technique that you'll no doubt have experienced in smartphones and tablets.

To pinch and zoom, first ensure that the zoom tool is selected at the top right of the screen:

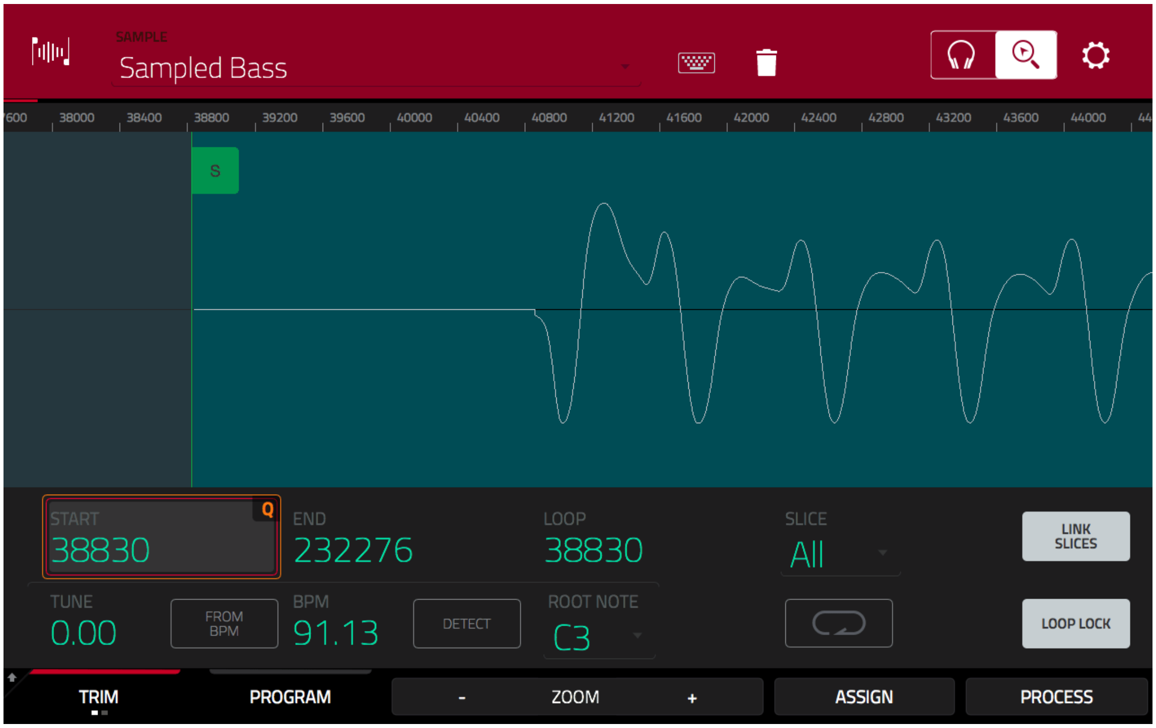
Now place both your index and middle finger together on the screen over the start of the wavelength (or some people prefer their index finger and thumb) and expand both or one of the fingers sideways. As you do, you'll see the waveform enlarge (if your waveform disappears off screen, just use your finger to drag the waveform to the left or right).



If you are not zoomed in enough, simply pinch and zoom again until you get the resolution you wish. You can also zoom in by turning the **bottom Q-LINK dial in the fourth Q-LINK column** clockwise or use the dedicated **ZOOM+** button.

You'll notice that the 'pinch and zoom' method is not currently able to magnify the waveform as much as the other methods, so if you want to get really close, use the bottom Q-Link in bank 4, or the **Zoom +** button.

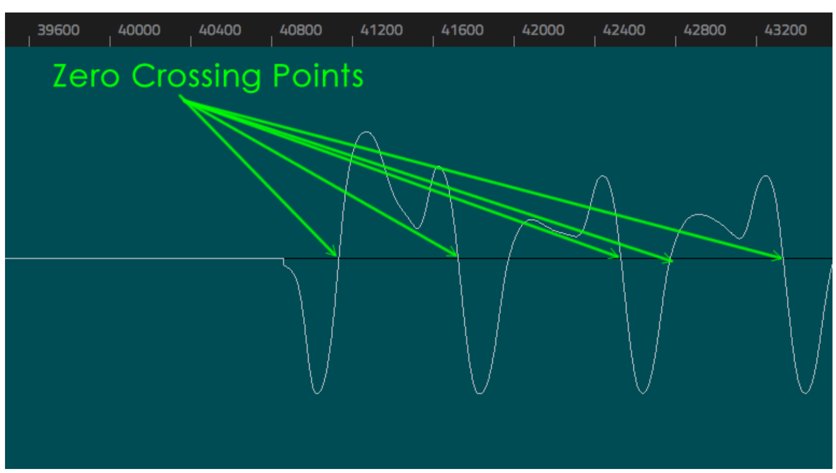
Try to zoom in so you can easily see the beginning of the waveform:



As you can see, we can still move that START point closer to the start of the waveform.

Snap to Zero

When editing waveforms it is normally best to place your edit points on 'zero crossing points'. This is simply the point where the waveform crosses the horizontal line running through the waveform:



In most cases this is the best place to put an edit point as it normally avoids any 'clicks' in the audio. To have the MPC *automatically* place your edit point to the nearest zero crossing point, we'll turn on **SNAP TO ZERO.** Hold down the **SHIFT** button and notice how the buttons at the bottom of the screen change:

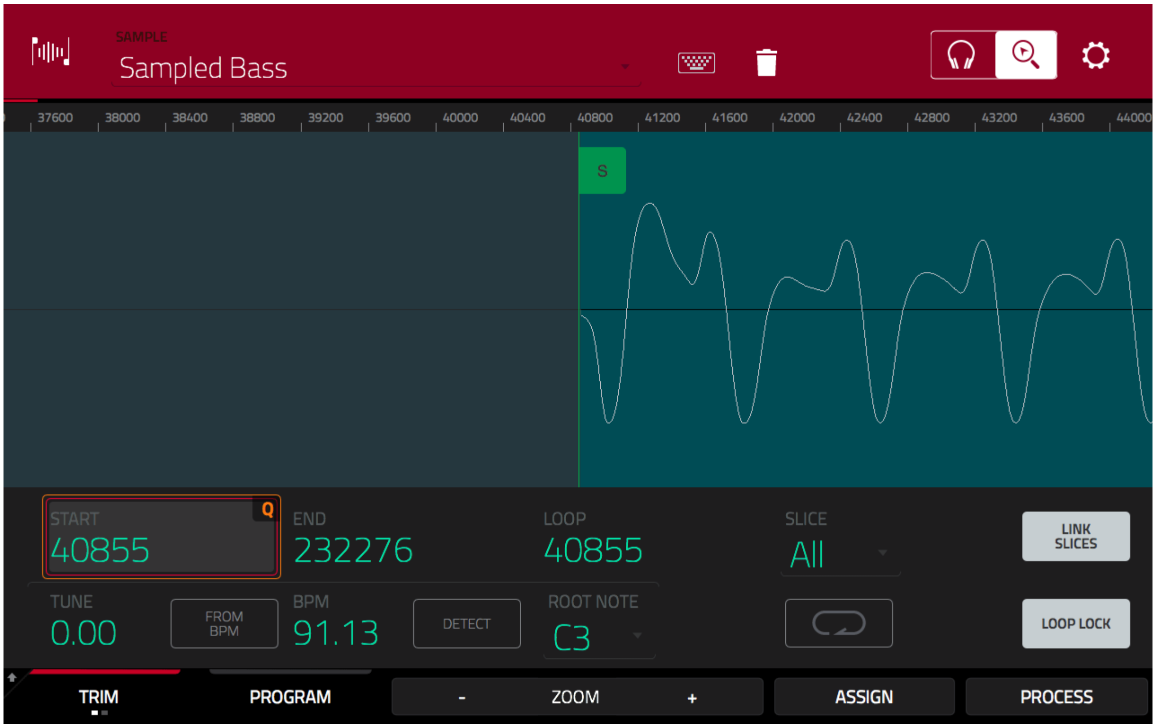
To turn on **SNAP TO ZERO**, simply press **0 SNAP** so the button turns grey with a red line at the top:

Setting the START Point

To move the START point, use any method you prefer. If you want to use the Q-LINKS, use the second from top dial to move the bulk of it and then finish up with the bottom Q-LINK. Alternatively tap on the START Point and turn the data wheel clockwise – the accuracy of each data wheel 'click' is proportionally linked to the current zoom magnification, so at highest magnifications each 'click' of the wheel represents a single sample point.

Just remember that with SNAP 0 selected, the START point will jump to each successive zero crossing point.

At maximum magnification, set your start point to **40855**:



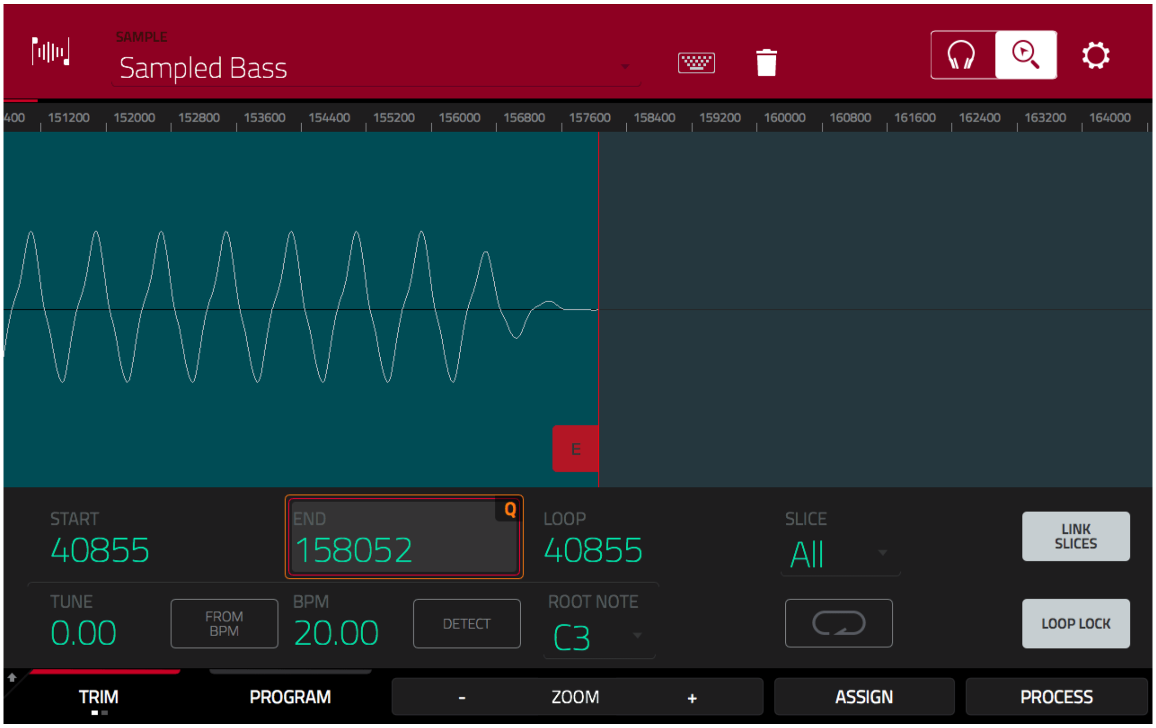
Preview the audio from the new start point using **pad 1** (Play From Start) or **pad 10** (Play Sample) – it sounds great.

Adjusting the End Point

The process for adjusting the end point is pretty much identical to that for the start point, except this time we want to remove the dead space at the end of the sample.

As before we can zoom in closer, drag the end point marker directly on screen, tap the END parameter and turn the data wheel, or use the Q-LINKS (this time it's **Q-Link column 2** for END point adjustment).

Using your preferred methods, set the **END** point to **158052**.

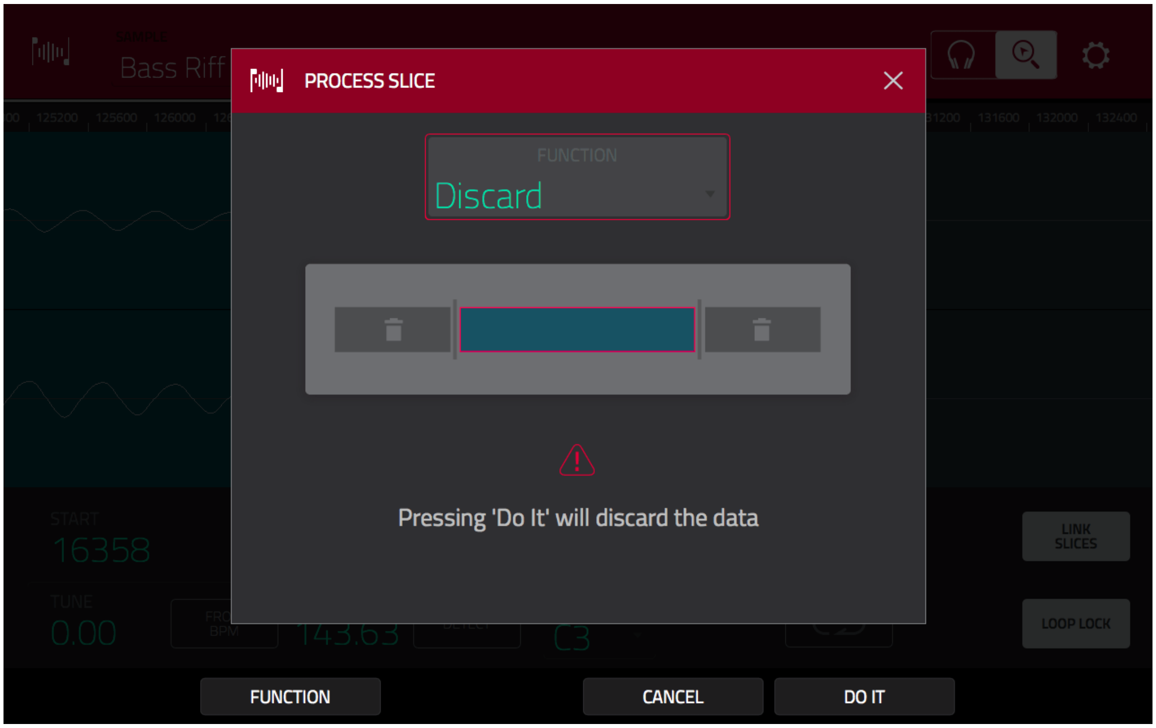


To hear what this sounds like, press pad **10**. That sounds pretty good – now press **pad 4** (**PLAY *FROM* END**) – this plays all the audio *after* your end point and hopefully you should hear absolutely nothing (you can check more accurately using headphones). All sounding good so far.

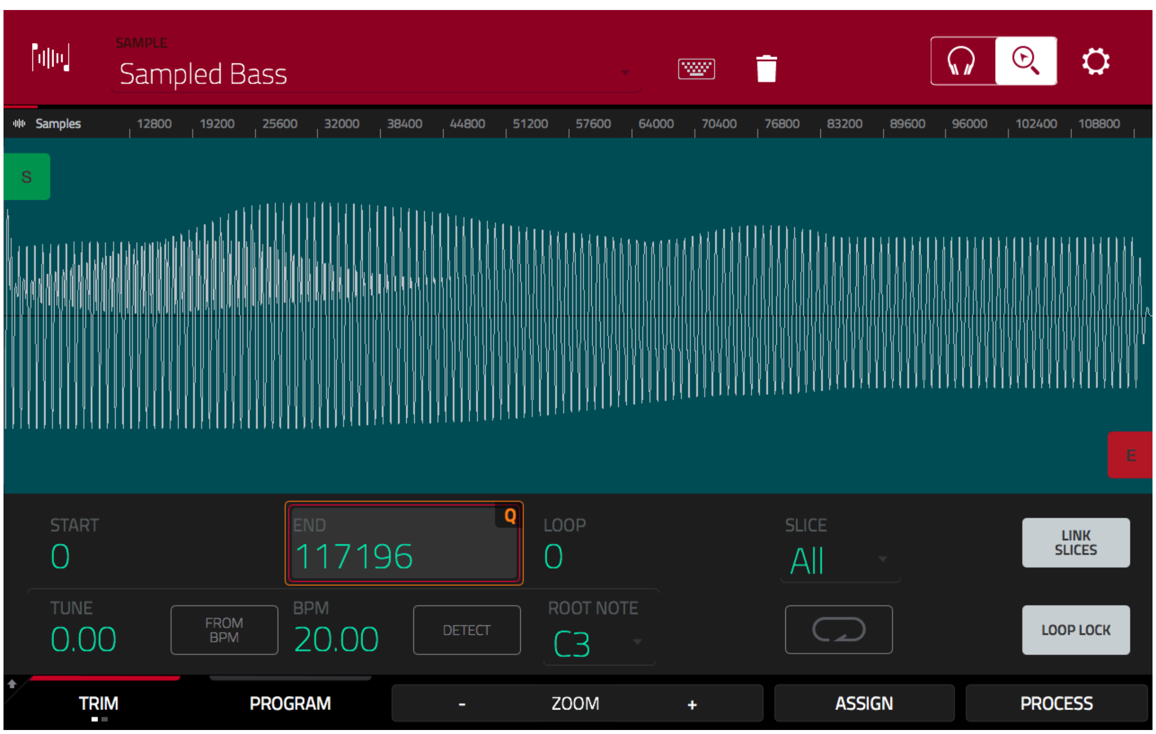
The Discard Function

We no longer need those areas of silence either side of the START and END points, so let's get rid of them!

Press the **PROCESS** button at the bottom right of the screen to bring up the '**Process Sample**' dialogue. With the **FUNCTION** parameter selected, use the data wheel to select the '**Discard**' function:



The DISCARD function simply removes all sample data outside of the start and end points that you previously set - any data in between these points shall be kept. To discard unwanted data, just press **DO IT**. You now have a perfectly trimmed sample that uses the minimum amount of storage and memory.



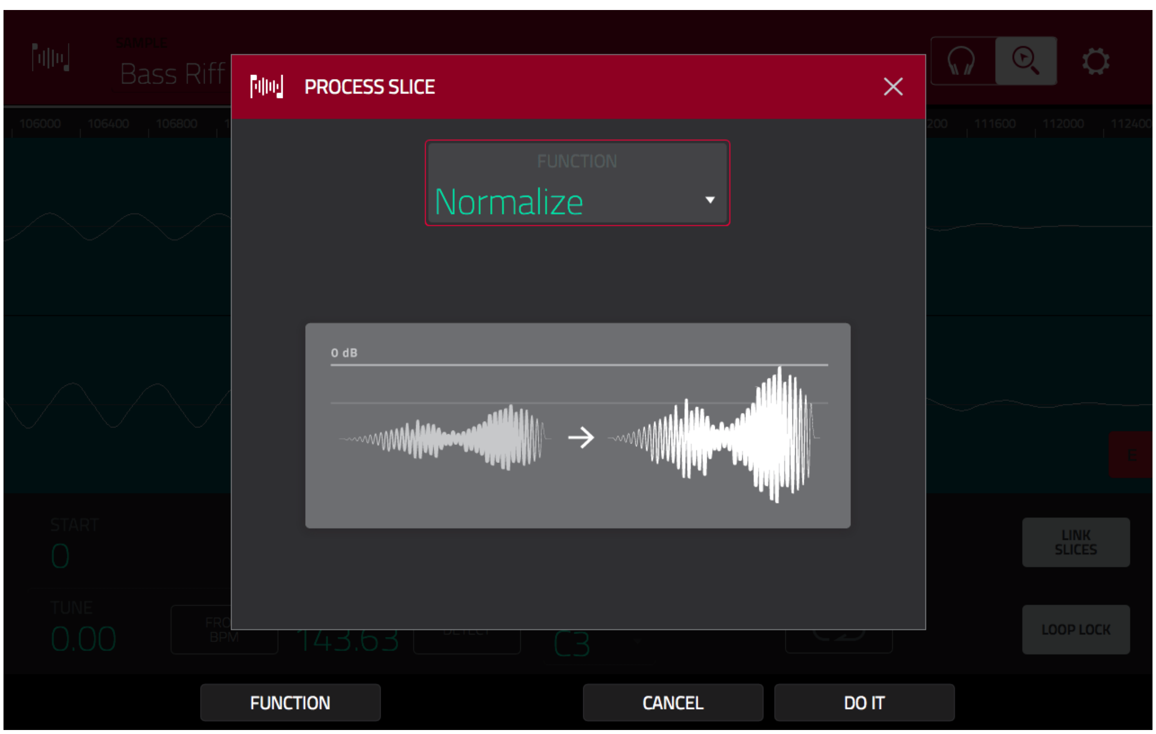
If you make a mistake, don't worry, as you can 'undo' the discard edit by simply hitting the dedicated **UNDO** button on your MPC hardware. If you decide you don't like the 'undo', hold down **SHIFT** and press **UNDO** again and it will '**redo'** the original edit (i.e. undo the undo!). If you repeatedly press the UNDO button you are able to undo up to 512 steps.

Normalizing

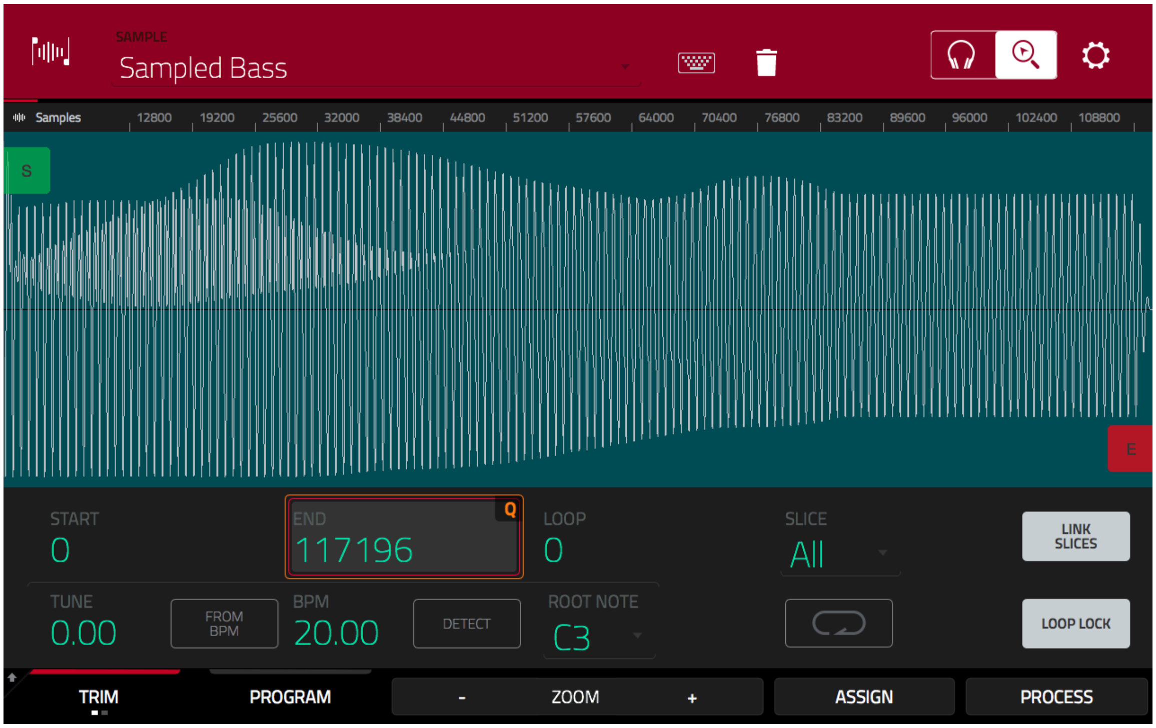
After trimming your sample's start and end points, one operation you may possibly consider is the process of **normalizing.** Normalizing will simply raise the gain of the *entire* sample so that the loudest peaks in the waveform sit at 0dB, giving the maximum output without making any actual changes to the make up of the file itself.

Normalization is no substitute for recording your sample at the hottest level possible in the first place, as it also increases the level of any existing 'noise present in the sample, but it is a great way of increasing the working volume of an already recorded sample.

To normalize your sample, simply press **PROCESS**, select **'Normalize'** and hit **Do IT.**



Your sample is now at maximum level:



We'll become more familiar with many of the other sample processing functions as we progress through the book.

Renaming an Edited File

You may have noticed there is a keyboard icon next to the sample name in SAMPLE EDIT; press this to rename the currently selected sample.

You can check out my version of the trimmed bass sample by loading up '**Dub Bass Note**' from the **008** folder.

009 Recording a Bass Line With 16 LEVELS

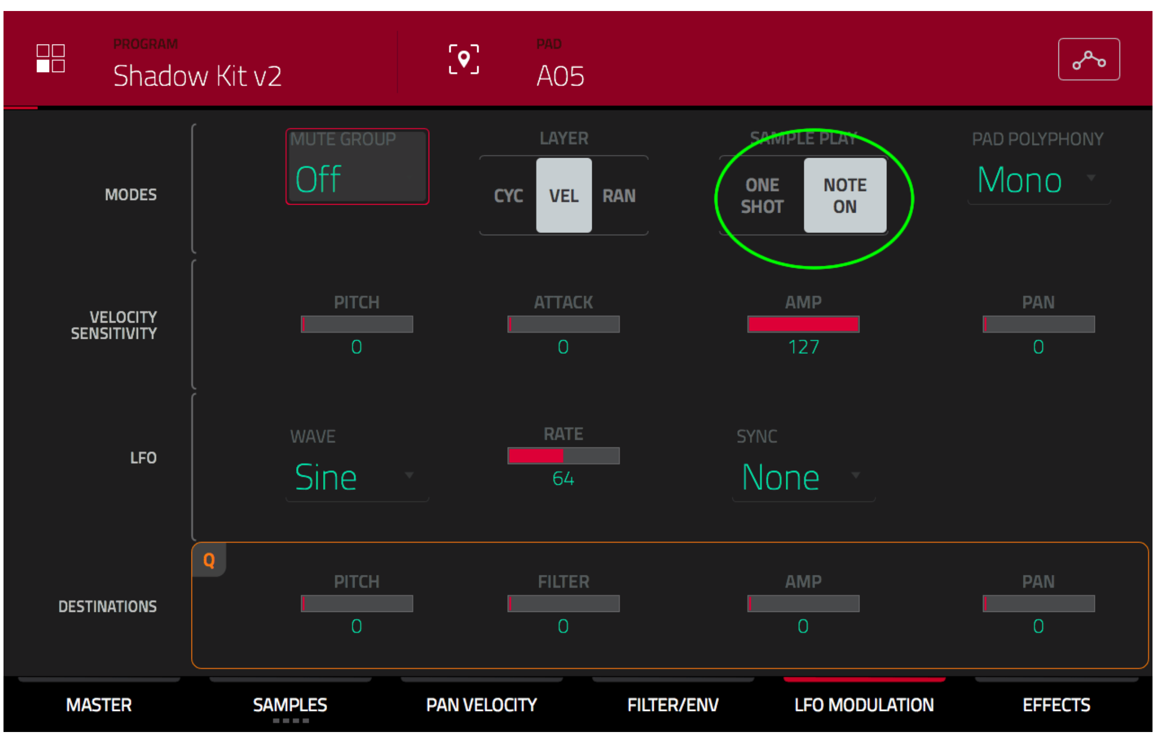
So, we've sampled and 'topped and tailed' a really nice bass note. Wouldn't it be great if we could use this one single note to create our own bass lines?

Setting Up The Bass Sample

Even if you only have a single bass sample, we can use that single sound to create a bass line for our beat. When we originally loaded the bass note from the previous sampling tutorial (**Dub Bass Note**), we simultaneously assigned it to pad A05.

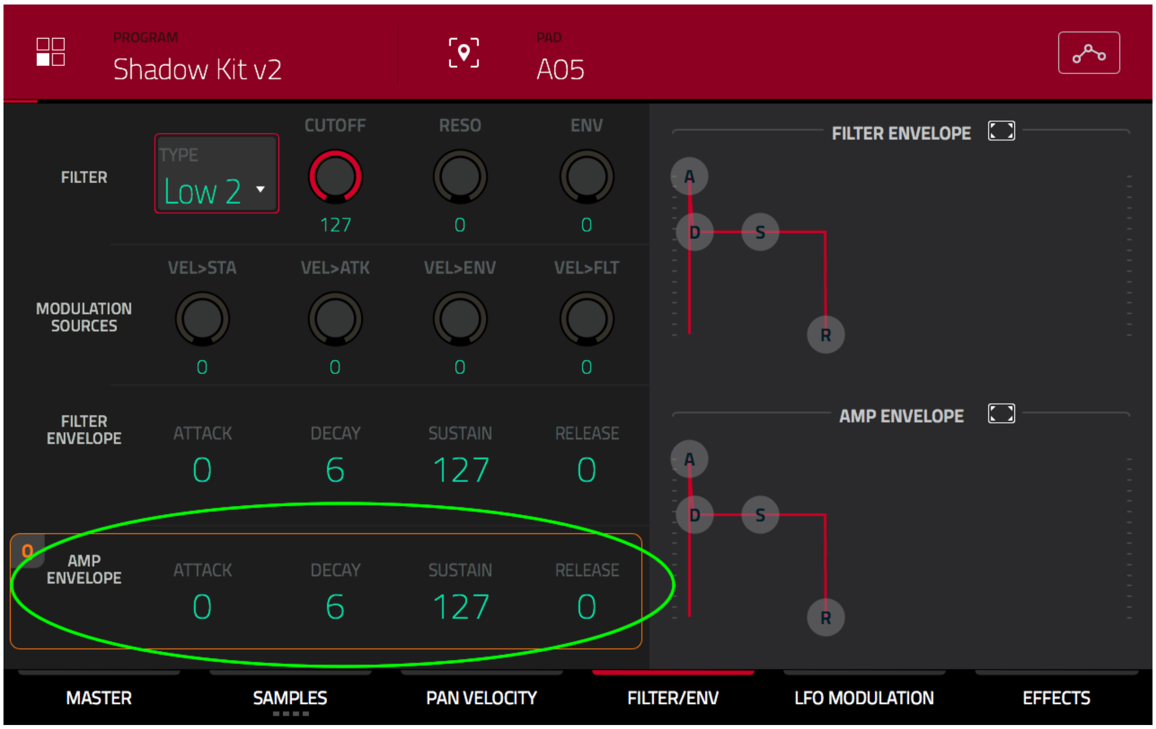
Preview pad **A05** – this bass note is very loud compared to the rest of the kit, so let's head over to **PAD MIXER** to adjust the volume of this pad. Set the volume for **A05** to **-8.80dB**.

Now currently when hitting A05, our bass note plays all the way through from start to finish. When playing an instrument note it would be better if we could dynamically control the length of the sound, just like we'd expect when playing a bass patch on a keyboard. Go to **PROGRAM EDIT > LFO MODULATION**. Change the **SAMPLE PLAY** setting from ONE SHOT to '**NOTE ON**'.



Now *briefly* touch pad **A05**, and you'll hear that the bass sound stops playing the moment you take your finger off the pad. That's much better, however when you let go of the pad you can hear a nasty click; to fix this we need to head back to our old friend the '**Amp Envelope'**.

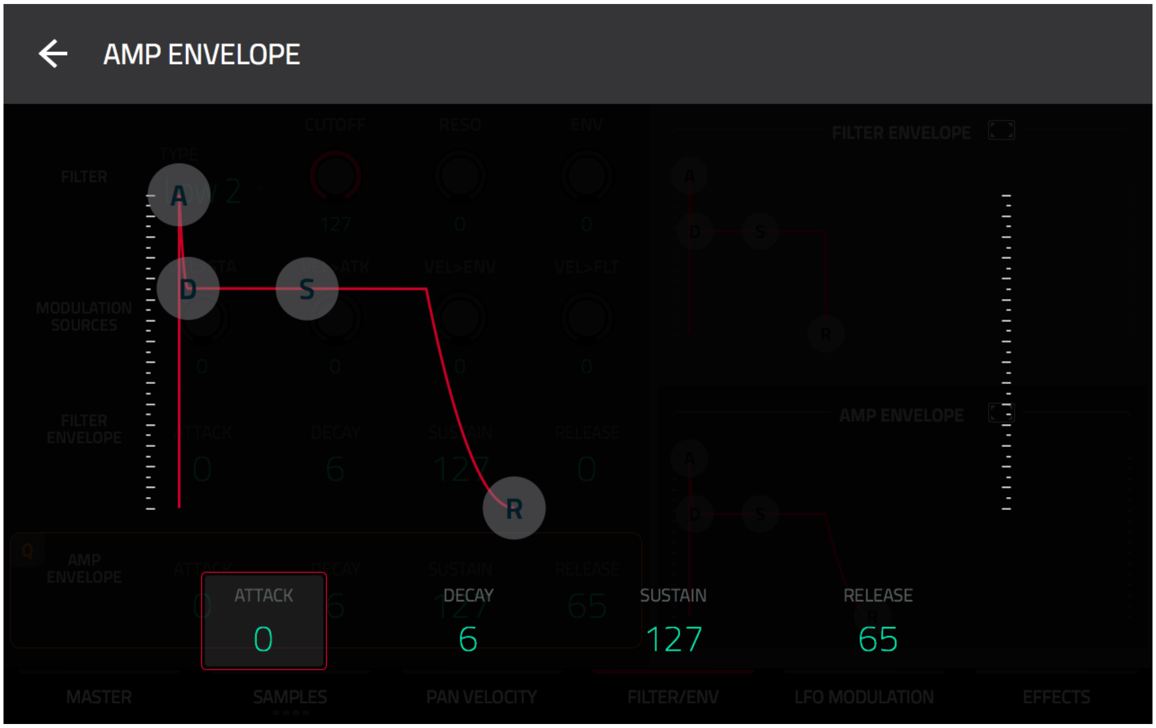
Press **FILTER ENV** and refer to the bottom '**Amp Envelope**' row.



Instead of the usual AD (attack and decay) envelope we see that selecting NOTE ON has given us an **ADSR** envelope with four adjustable parameters. ADSR stands for '**a**ttack, **d**ecay, **s**ustain and **r**elease'. Take a look at an example envelope:

The attack (**A**) works in the exact same manner as the AD envelope, the **decay (D)** now controls how long it takes for the envelope to reach the **sustain** portion (**S**) which, as you'd expect, is a portion of the envelope that is 'sustained' at a constant level. Finally the **release** (**R**) controls the time it takes for the sample to fade to nothing after you physically release the pad.

In this example, let's just concern ourselves with the '**Release'** setting at the end. With a default release of zero, when you stop playing the sample, no faded decay is added, and hence you get that nasty sudden click at the end. But the higher the release value you add, the more gradual decay is added to the end of the sample. Keep briefly previewing pad **A05** as you gradually adjust the '**RELEASE**' value. You should gradually hear that click at the end disappear at around **40**, but let's set it to **65** to also give the bass note a little bit of additional fade at the end so it sounds more natural.

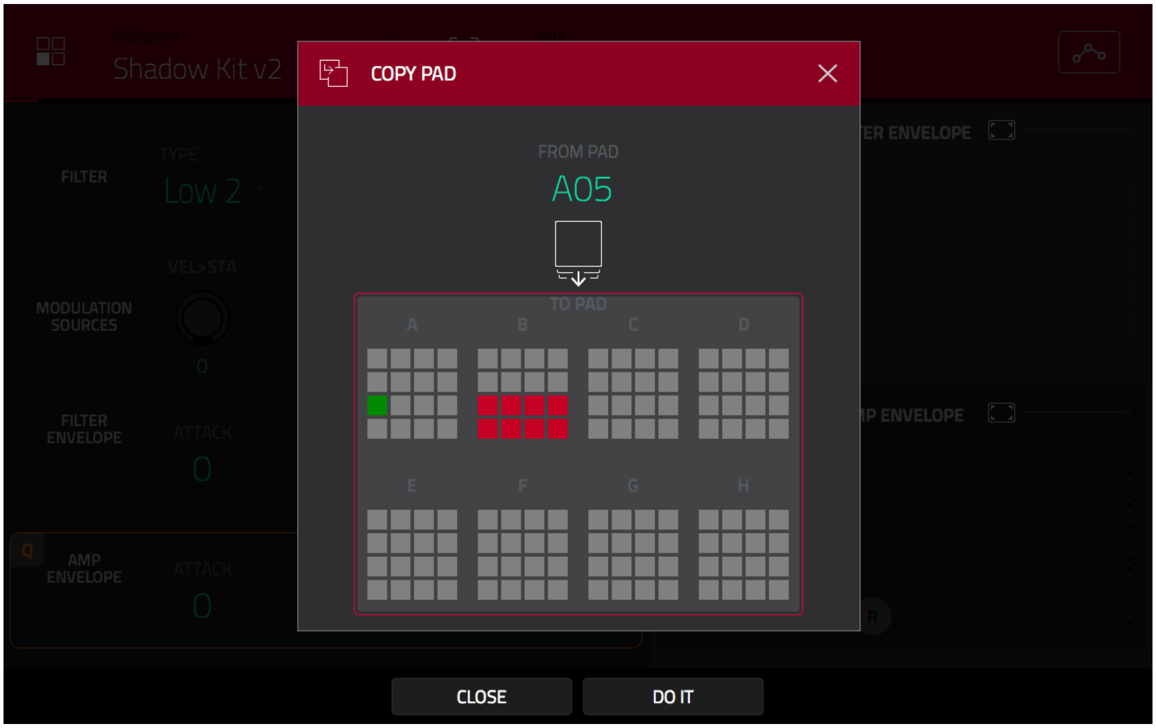


Changing the Pitch of your Sample

Now, we can't really make much of a bass line with a single bass note, but it's very easy to create some extra notes from this one bass sample. The distance between a musical note in the standard musical scale (C, C#, D, E etc) is called a 'semitone', and in the MPC this difference is represented by a '**SEMI'** tuning change of **+/- 1**.

Hold down the **COPY** button to bring up the **COPY PAD** screen, and while still holding down the COPY button, hit pad **A05**. This will set A05 as the '**FROM PAD**' and simultaneously enable the ability to add some '**TO'** pads.

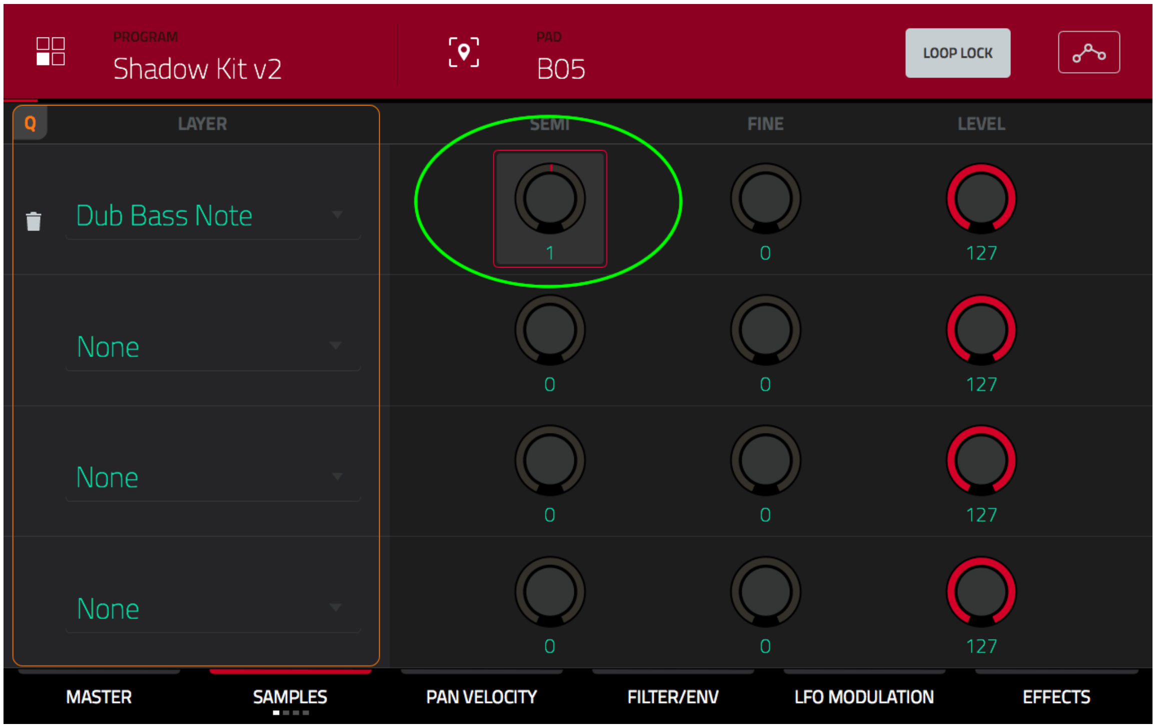
You can now hit any pads you wish to copy pad **A05** to – select **BANK B** and hit pads **B01** to **B08**



Once you have selected them all, hit **DO IT.** Now preview all eight pads and you'll hear they are all identical.

The next stage is to begin tuning all these pads to create a 'chromatic' run of notes. Chromatic simply refers to the standard semitone change you get when running through the standard musical notes.

So, let's use pad **B04** as our '**0**' tune pad – i.e we'll leave this pad tuned to the default '0'. Now select pad **B05** – in the **PROGRAM EDIT > SAMPLES** screen, you'll need to change the **SEMI** value for the top layer to '**1**'.



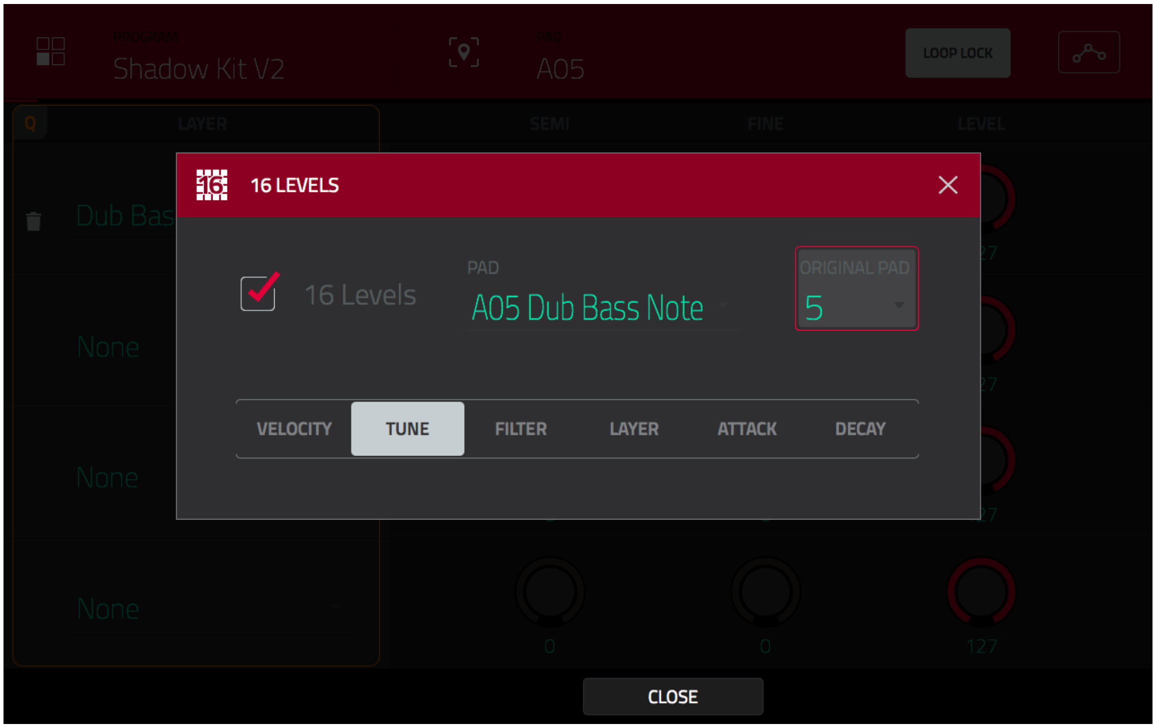
Now press pad **B06** and press the **+** button twice to increase the **Semi** tuning to **2**. Set the **Semi** value of **B07** to '**+3**' and **B08** to **+4**. Now we'll drop the pitch for the pads that come before our 'zero' pad B04. So set the **SEMI** for **B03** to **-1**, **B02** to **-2**, **B01** to **-3**.

Now starting on **B01** press all the pads one at a time in order to hear your chromatic bass guitar emulation. It's clear it would be easy to create a bass line from these eight pads and you could of course continue this across all the pads in this bank.

However, there is a far quicker way of mapping a bass note across all the pads - 16 LEVELS.

Recording a Bass Line with 16 LEVELS

Head over to **BANK A** and select pad **A05**. Now press the dedicated **16 LEVEL** button in the hardware. Make sure '**TUNE**' is selected so the '**ORIGINAL PAD**' option is activated. Set '**ORIGINAL PAD**' to be '**5**'.



Hit '**CLOSE**' and now run through pads **A01** to **A16**. The first thing you might notice is that our drum kit has disappeared – instead we have our bass note mapped out chromatically across all 16 pads in BANK A.

In 16 LEVELS mode, all 16 pads are really just active clones of the same 'master' pad (A05). With **TUNE** selected as the '**TYPE'**, each pad features with a progressive tuning change of +/- 1. So when you play pad A06, in reality you are simply playing pad A05 with a +1 tuning. Hit pad **A04** and you get pad **A05** played at a tuning of **-1**. Hit pad **A16** and you'll get pad **A05** with a **TUNE** of +**11**.

All other existing samples on the pads in your program are ignored/hidden while you are operating in 16 LEVELS mode – only the 'master' pad is used (in this case, A05). Don't worry, your drum kit will return as soon as you leave 16 LEVELS mode (easily done by pressing the 16 LEVELS button again).

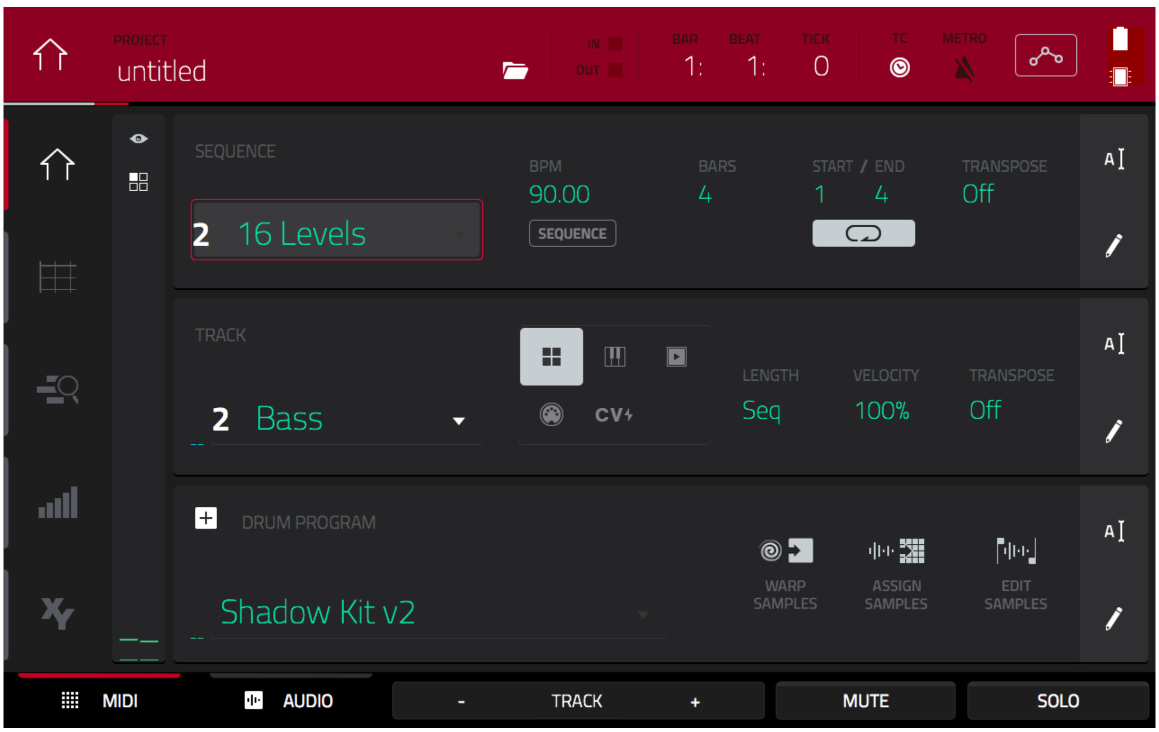
So with these tuned bass notes temporarily mapped across our pads, it's time to actually record our bass line!

Configuring Your Sequence

Load up the **16 Levels.sqx** sequence from the **009 folder** – this is a 4 bar beat recorded on track 1 based on the sequences we created during the Grid Edit and Quantising chapters. Press **PLAY START** to hear it.

We've now got four bars of drums to record our new bass line over. Now we could record this bass line to the same track as our drums, but generally speaking it's normally a good idea to place different 'instruments' on their own tracks. Why? Well apart from being a much more organised way of working it can make mixing significantly easier; you'll also find 'tracking out' your beat to third party software (e.g. DAWs) a far more pleasurable experience. We'll look at these issues in more detail later in the book.

So let's create a dedicated sequencer track for our bass line. In **MAIN**, tap on **TRACK 1 Drums** and hit the **+ button** – alternatively, just hit the **TRACK +** button at the bottom of the screen. Your new track is now created and is currently labelled as 'unused', so hit the '**A**' icon at the end of the row and rename it to **Bass**.



You're now ready to record your bass line to track 2 of your current sequence. Initially press **PLAY START** and play around with some bass line ideas. Use any **TC** setting you prefer – the same goes for **FULL LEVEL**, turn this on if you wish. I'd actually advise that you do as this is a big and deep bass sound so any variances in velocity will be quite obvious and might detract from the rest of the beat.

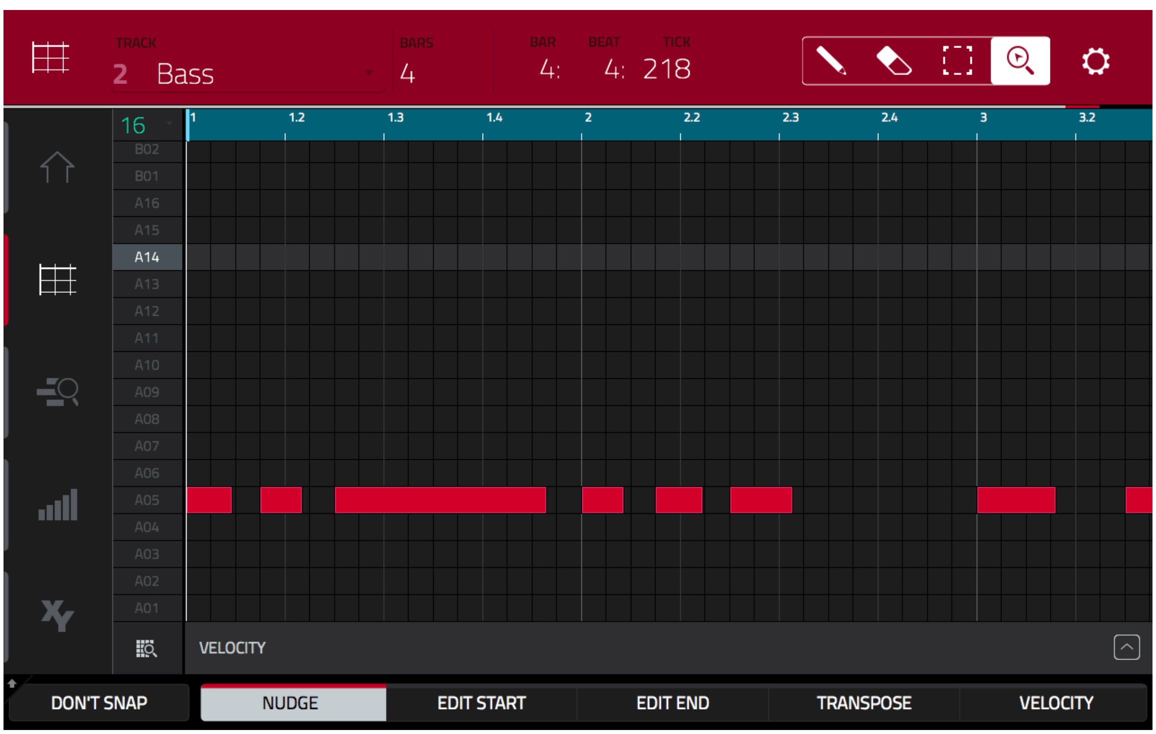
To record your bass line, hit **REC** and **PLAY START** and start hitting your pads!

Changing Program Parameters In 16 LEVELS mode

If you wish to change any program parameters to the sound you've applied 16 Levels to, you must make the changes specifically to the 'master' pad (in this case pad A05) and make the changes *specifically* to that pad. Any changes to that master pad are then automatically reflected in all other pads in bank A, because they are all simply *virtual copies* of pad A05 with a different tuning applied.

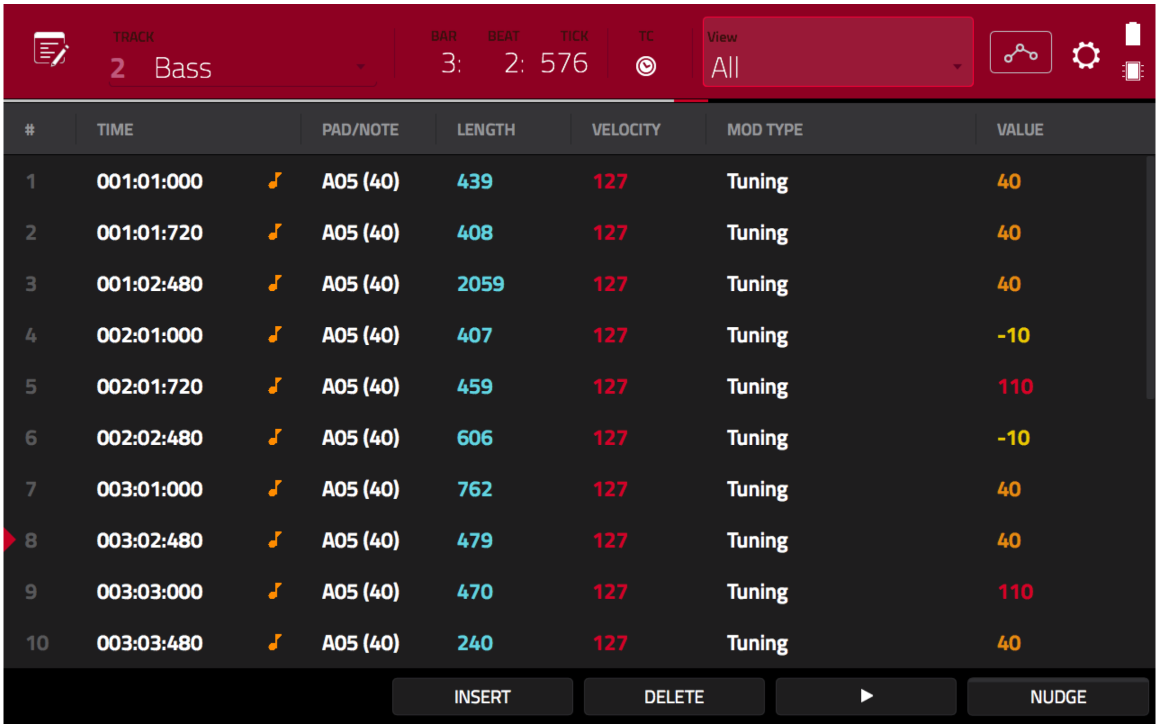
LIST EDIT Mode

Load up the sequence **16 Levels Bassline.sqx** to hear my bass line. Feel free to turn off 16 LEVELS – it doesn't need to be 'on' to play back previously recorded 16 levels performances – it only needs to be engaged during the actual recording process. This is because in 16 LEVELS mode, all the tuning changes are instead stored within the sequence itself, not the program. Take a look at **track 2** in **GRID EDIT**



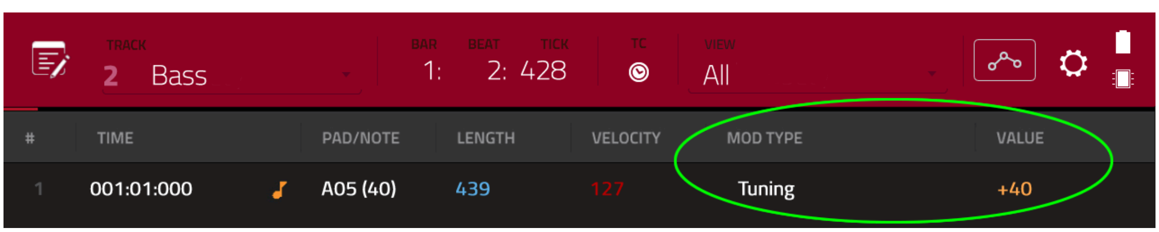
As you can see, even though we can hear lots of different notes during sequence playback, the sequence itself only contains multiple instances of pad A05. So what's going on here?

The answer is revealed in a separate sequencing mode called **LIST EDIT**. To enter LIST EDIT, go to **MENU > LIST EDIT** or hit **SHIFT > STEP SEQ** in the MPC X.



LIST EDIT is a purely text-based representation of the currently selected sequence track where each MIDI event in the track is displayed as a series of parameters.

Take a look at the 'step 1', **TIME 001:01:000**



Here we have a 'note' event (as indicated by the yellow note icon). The value of the note is displayed in the **PAD/NOTE** column; **A05(40)**. This indicates that this is a **pad A05** note event (the 40 refers to the MIDI note number assigned to this pad, we'll look at MIDI note values in detail as we progress through the book).

The **LENGTH** column displays the length or duration of the note event (**439**). The **VELOCITY** column indicates how hard the event is to be played (**127**).

The final two columns are the key to 16 LEVELS; **MOD TYPE** and **VALUE**. **MOD TYPE** is short for '*modifier type*' and as we can see, is set to **Tuning**, with a **VALUE** of **+40**. The MPC has 'modified' our sample by tuning it to +40, i.e. it has raised its pitch by 4 semitones.

It's a similar story for the rest of the 16 LEVEL bass note events in this track, with the 'Tuning' value varying to reflect the different pitch played when in 16 LEVEL mode; for example, the note at **002:01:720** (**step 5**) has a '**Tuning**' of **+110** (a pitch increase of 11 semitones).

So, in 16 LEVEL mode, while we are always recording the *same* 'note' (pad A05), we apply a unique tuning modification to each note depending on which '16 level' pad we hit.

Inserting Events in LIST EDIT

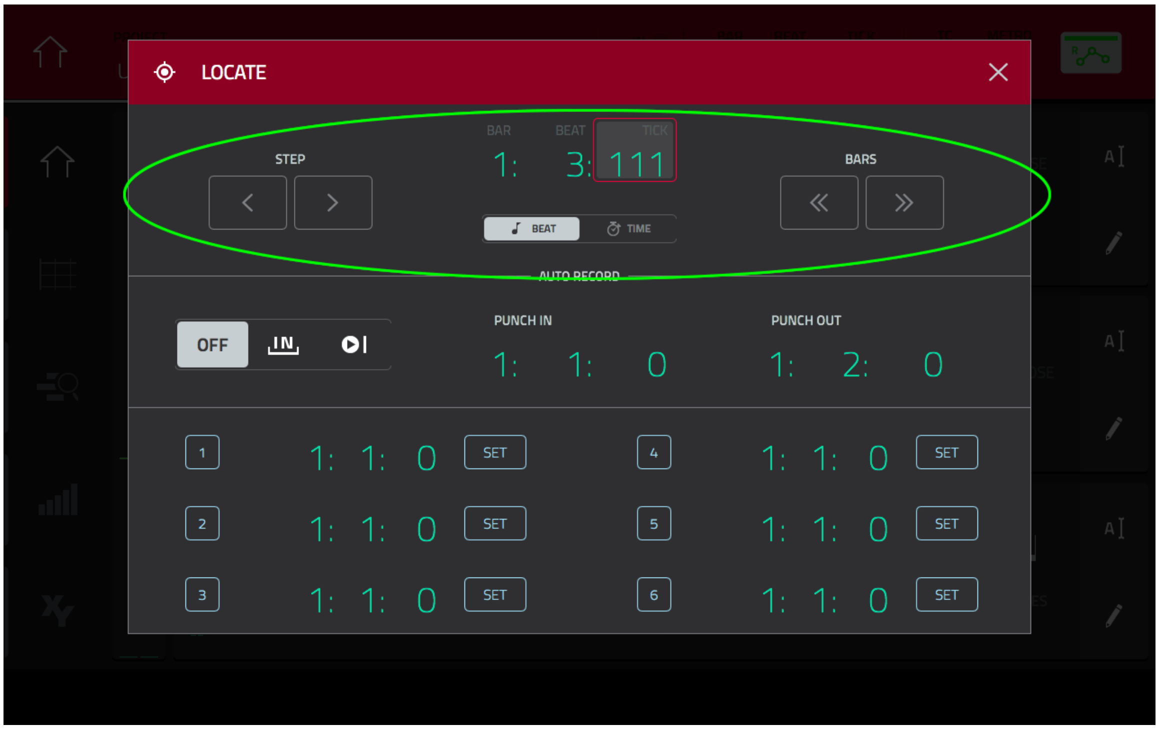
Let's insert some additional events to this sequence. First, make sure you have a **T.C.** of **1/16** - to set this in LIST EDIT you'll have to hit the **TC** icon at the top middle of the screen.

Now go to time location **001:03:000** - let's take a look at the various ways we can do this.

In the MPC X you can locate to different time locations using the **STEP** and **BAR** buttons:

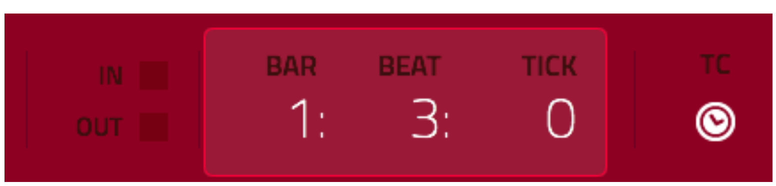
Each tap of a STEP button will move you to the next TC quantise point. Each tap of a BAR button will take you to the next bar.

There are no physical equivalents of these buttons in the MPC Live, but there are software versions. In LIST EDIT, double tap the time locator at the top middle of the screen to bring up the **LOCATE** screen (MPC X owners can hit the **LOCATE** button on their hardware, it's just above the STOP button):



Here you can use the on-screen **STEP** and **BARS** buttons in the same way as the physical MPC X versions. You can also edit each **BAR/BEAT/TICK** parameter in the time display(single tap and turn data wheel or double tap and enter the number).

Alternatively, you can adjust the time locator in the LIST EDIT screen:



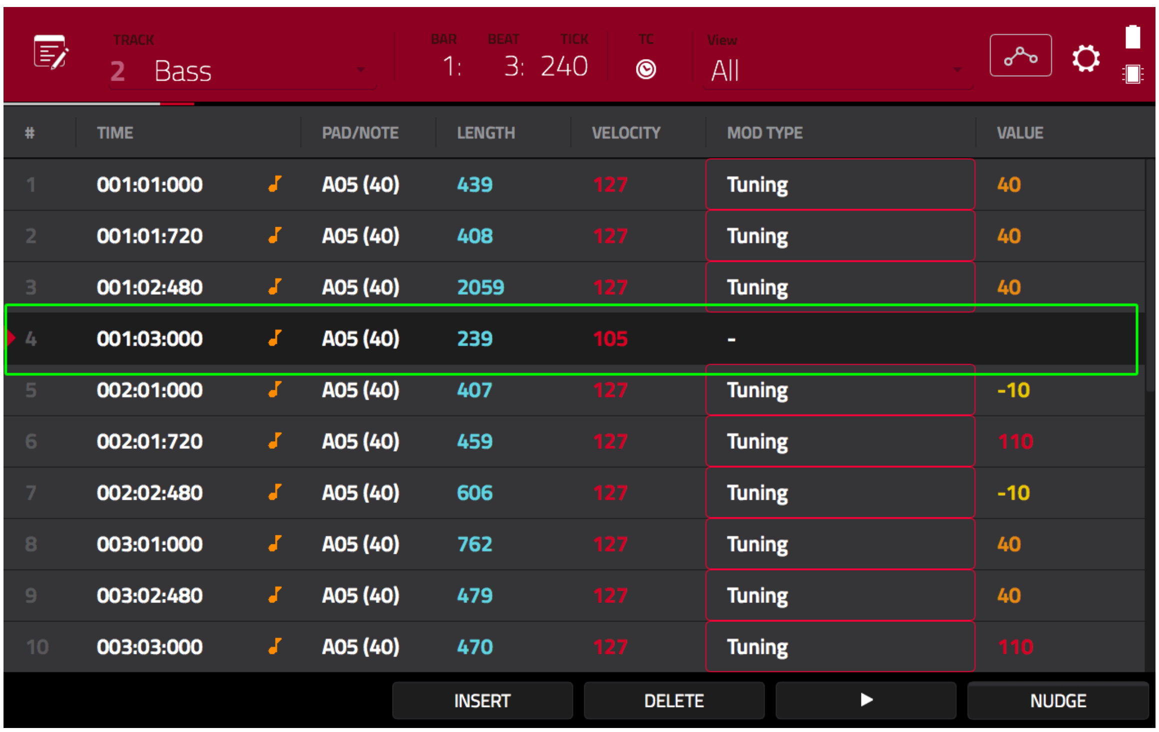
Just single tap the entire time locator and turn the data wheel to move one 'beat' at a time (regardless of the current T.C. value). However, if you hold down **SHIFT**, each data wheel 'click' will move the time locator by a single **T.C**. value.

To '**locate**' to time **001:03:000** use any method you wish (in the MPC Live I'd personally just tap the time locator and turn the data wheel to move in beats) - ensure the display reads **1:3:0**.

There are actually two ways you can insert a new pad A05 '16 LEVEL' bass note event here.

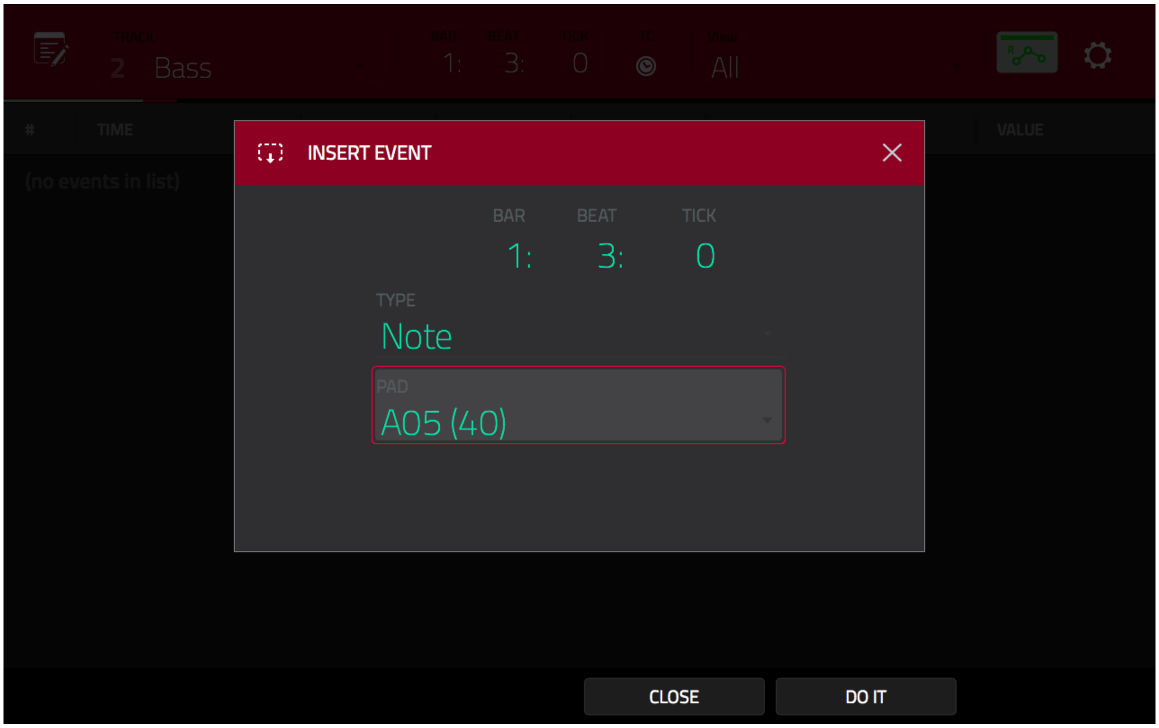
Method 1: OVERDUB

Press the **OVERDUB** button to put LIST EDIT into 'record-ready mode' and hit pad **A05**. You should how see a new event appear at **001:03:000**:



Method 2: INSERT button

Alternatively, hit the **INSERT** button at the bottom of the screen:

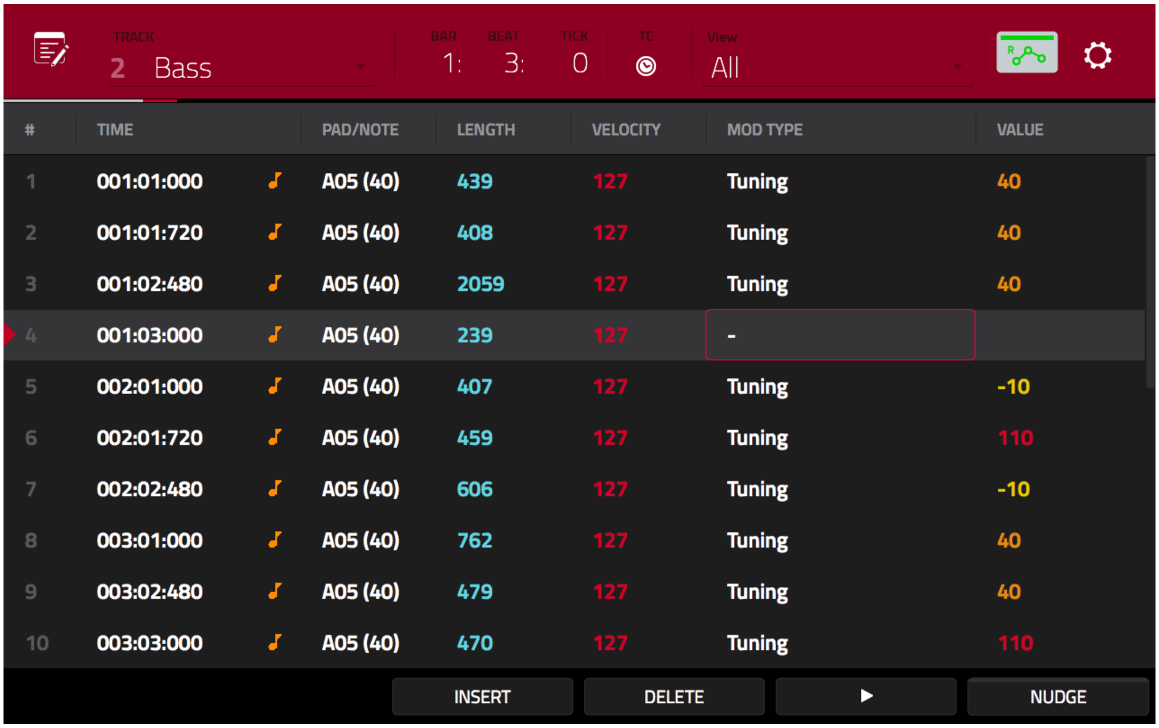


Make sure the time location is correct, then choose **TYPE: Note** and **PAD A05(40).** Hit **DO IT** - the result is the same, an **A05** event at **001:03:000**.

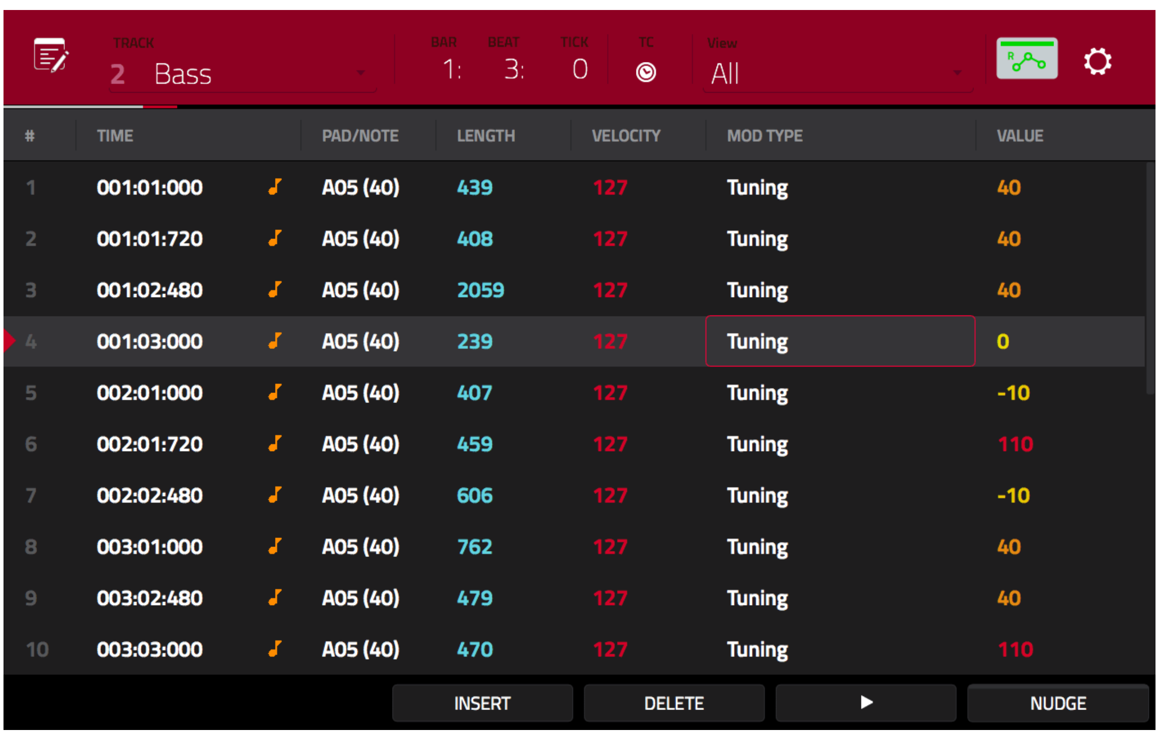
Editing Note Events In LIST EDIT

Currently our inserted event is jut a plain old A05 note event, so lets change this to a '16 LEVEL' 'tune' event.

In the 001:03:000 event (event step 4), single tap on the **hyphen** to select the **MOD TYPE** for our new event:



Now turn your data wheel one step clockwise so the **MOD TYPE** displays **TUNING**:



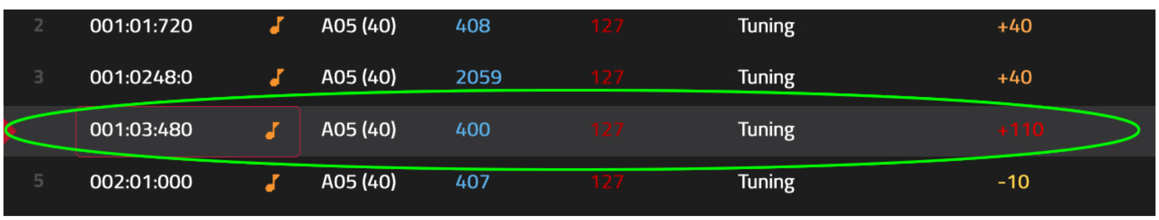
You can preview how this note currently sounds by hitting the **triangle 'play' button** at the bottom of the screen. This will play the event *exactly* as it has been recorded in the LIST EDITOR, i.e. the same note, pitch, velocity and duration.

Let's change the pitch of this inserted note. Double tap on the **VALUE** parameter (currently **+0**) and enter **110** via the on-screen numeric pad. As each semitone is a change of +/-10, this increase of +110 equals an increase of 11 semitones on the original note. Hit the **triangle play button** to hear how the note sounds and then hit **PLAY START** to hear it within the entire sequence

I reckon the bass note is a tiny bit too short, so tap on the blue **LENGTH** parameter (currently **239**) and turn the data wheel until it reads **400** (you can alternatively double tap to enter the number manually; you can also hold down **SHIFT** while turning to move in 'T.C' units, currently 240 units at a time). Preview the entire sequence via **PLAY START**.

Moving Events In LIST EDIT

Let's now *move* this new event. Tap on the TIME value for our event so it becomes selected and press the **NUDGE** button at the bottom of the screen. Now while holding down **SHIFT**, hit the **+ button twice** (or turn the data wheel two clicks clockwise) to move this event **'2' quantise points** to **01:03:480**:



2	001:01:720		A05 (40)	408	127	Tuning	+40
3	001:0248:0		A05 (40)	2059	127	Tuning	+40
	001:03:480		A05 (40)	400	127	Tuning	+110
5	002:01:000		A05 (40)	407	127	Tuning	-10

(If you don't hold down SHIFT the TC value is ignored and the note will just move 2 ticks to 001:03:002).

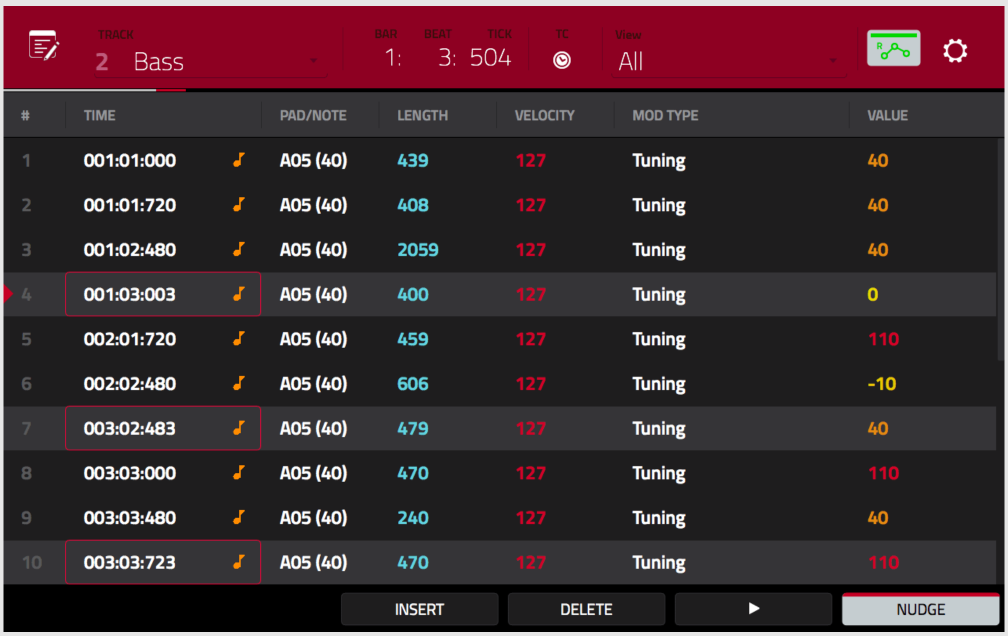
Remember you can of course continue to overdub any type of event into your track 'in real time' using OVERDUB + PLAY - if you make a mistake you can then use the LIST EDITOR to fix it or just just GRID EDIT.

If you wish to delete any event in your track, just tap to select it and hit the **DELETE** button at the bottom of the screen.

Moving and DELETING Multiple Events

You can edit more than one event simultaneously - just hold down **SHIFT** and tap all the events you wish to edit. To quickly select a range of notes, tap the first event, hold down **SHIFT** and turn the data wheel.

Once selected, press **NUDGE** and use the data wheel to change the **TIME** value as described previously - all the selected notes will move together. In the example below I've moved the three highlighted events 3 ticks forward.



#	TIME	PAD/NOTE	LENGTH	VELOCITY	MOD TYPE	VALUE
1	001:01:000	A05 (40)	439	127	Tuning	40
2	001:01:720	A05 (40)	408	127	Tuning	40
3	001:02:480	A05 (40)	2059	127	Tuning	40
4	001:03:003	A05 (40)	400	127	Tuning	0
5	002:01:720	A05 (40)	459	127	Tuning	110
6	002:02:480	A05 (40)	606	127	Tuning	-10
7	003:02:483	A05 (40)	479	127	Tuning	40
8	003:03:000	A05 (40)	470	127	Tuning	110
9	003:03:480	A05 (40)	240	127	Tuning	40
10	003:03:723	A05 (40)	470	127	Tuning	110

If you hit **DELETE**, all selected events will be deleted together.

You can load up my version of the 'list edited' sequence from the chapter 9 folder, '**16 Levels List Edit.sqx**'.

LIST EDIT vs GRID EDIT

The obvious benefit of LIST EDIT is the ability to edit parameters that are simply not available to your in GRID EDIT mode, such as the 16 LEVEL data we've been working with in this section, as well as 'automation' data (we'll be using automation in later chapters). It also allows you to perform very accurate tweaks of parameters such as LENGTH and VELOCITY, something that, while not impossible, is still very tricky to do in GRID EDIT.

Another advantage of LIST EDIT is it allows you to move events in single 'ticks' - GRID EDIT is limited to moving events in units of '4 ticks' at a time.

Additionally if you come from a 'legacy MPC' background, you'll probably feel a bit more comfortable working with this kind of numerical sequencer data as most of these older MPCs did not have any type of GRID mode at all (in those days LIST EDIT was referred to as STEP EDIT).

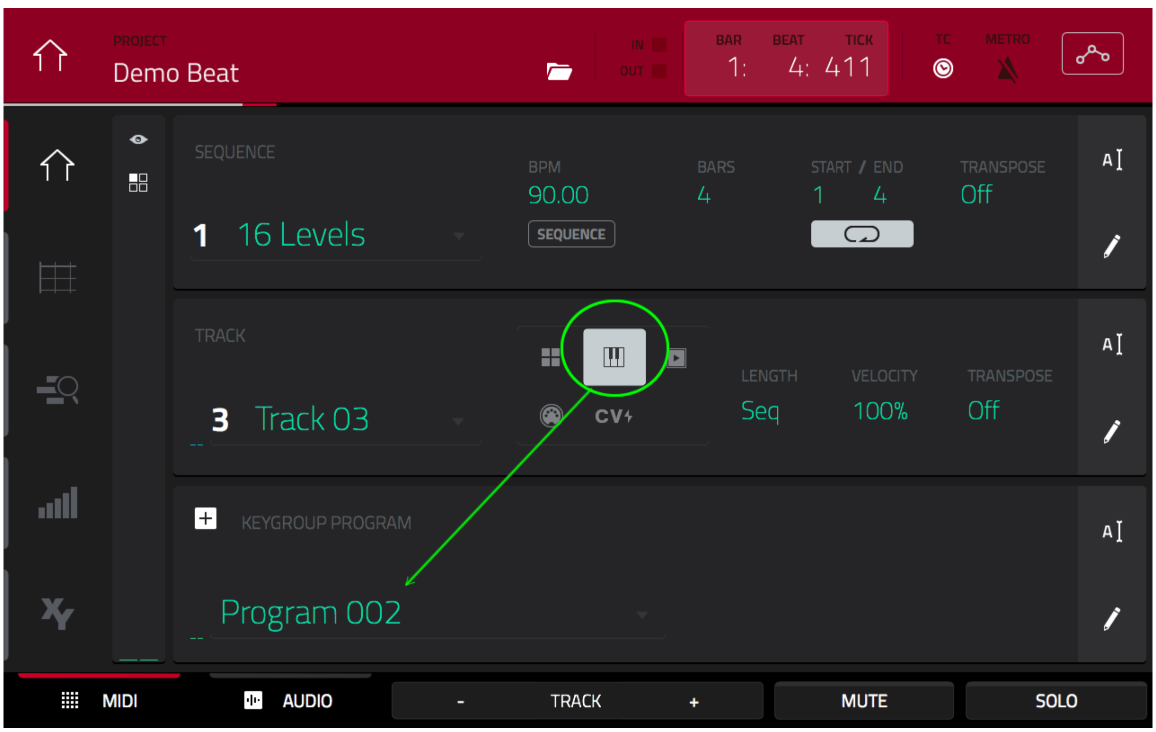
Ultimately it comes down to choosing the environment that works best for your needs. For me I'll often begin recording my sequences 'live' (usually in MAIN), then tweak some of the data in GRID and finally perform more detailed edits in LIST EDIT.

010 Introduction to Keygroup Programs

As we've seen, the DRUM program is very capable and flexible, but in this tutorial we're going to take a look at a program type that was created specifically with melodic instruments in mind; the ***keygroup*** *program.*

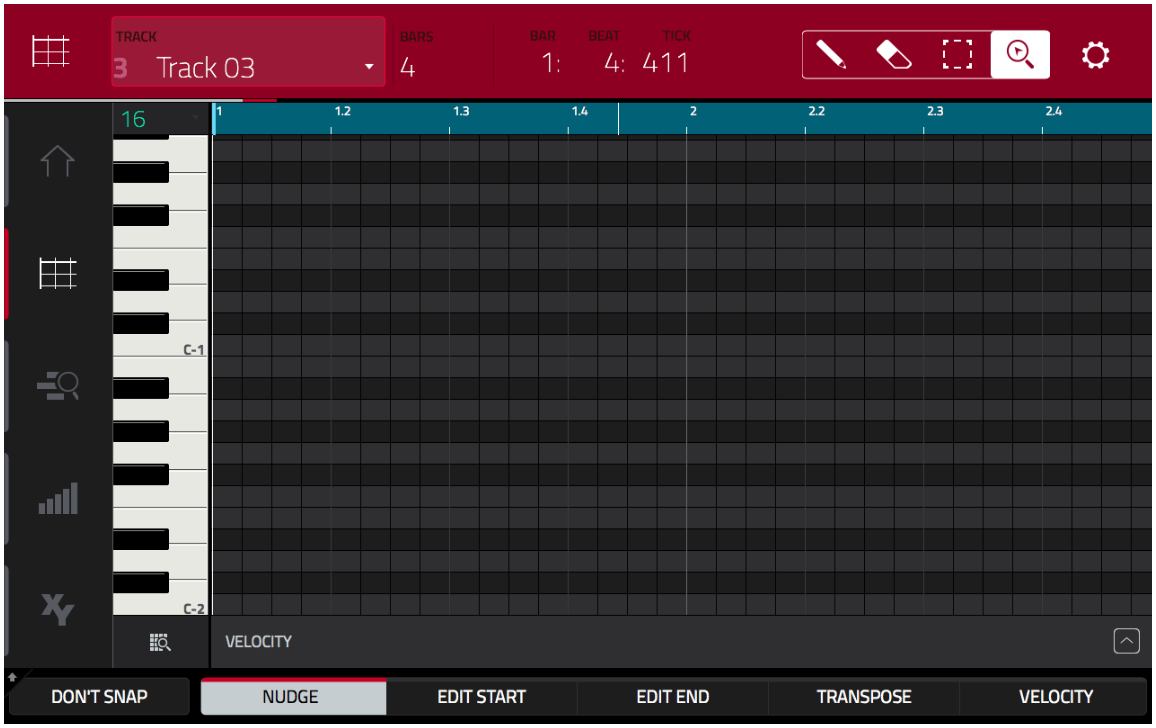
Creating Keygroup Programs

In your current sequence, select **track 3** (unused) using the **TRACK +** button. On the **TRACK** line, press the little keyboard icon.



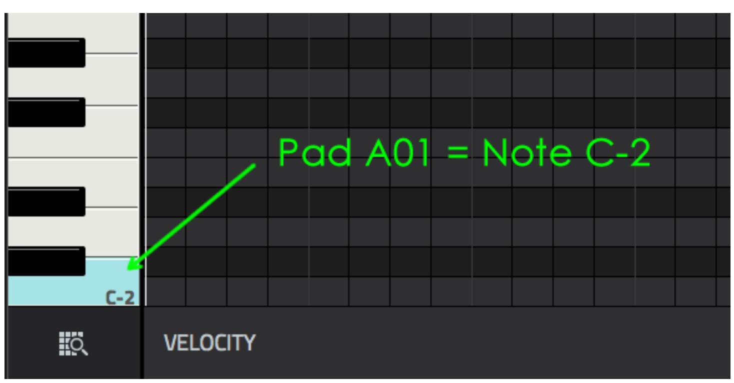
This has changed the track 'type' from DRUM to '**KEYGROUP'** and has simultaneously created a new KEYGROUP' type program (**Program 002**) and assigned it to the track. Rename this program to '**Piano Program**'.

Now there's plenty of differences between a KEYGROUP and DRUM program, but we can look at those as we move forward. Go to **GRID VIEW** and notice that rather than pad numbers running up the left hand side of the grid, we have a keyboard.



This is because a keygroup program is moving away from the general concept of ‘drum pads’ and is instead looking at the concept of ‘musical notes’ mapped out across a keyboard – in fact when working with keygroup programs it is best to now consider your MPC pads as nothing more than the ‘rubber keys’ of a virtual keyboard that will allow us to play musical notes.

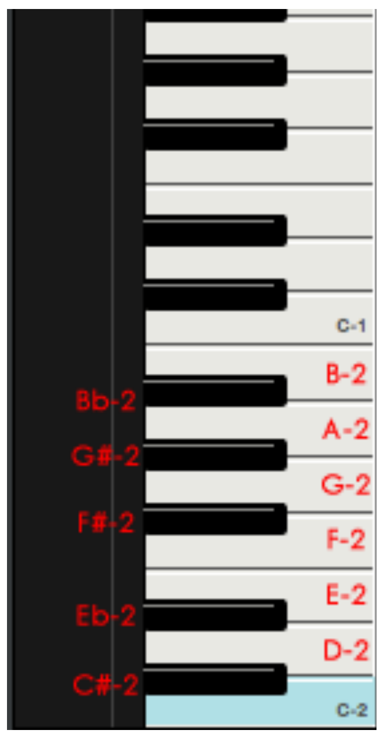
To highlight this difference, select **BANK A** and press some of the pads while you observe the on screen keyboard in the GRID. When you press pad **A01**, you’ll see that the (slightly) darkened row in the grid is in line with the key labelled **C-2** on the keyboard.



Press pad **A13** and you’ll see the darkened row is in line with the key referenced as **C-1**.

C-1 and C-2 are references to the musical notes that these two pads represent respectively. When we switch to using keygroup programs, we find that each pad has been assigned a **musical note**. As we've already seen, while we are in bank A, pressing pad 1 highlights the **C-2** note on the keyboard representation in the MPC Software. **C-2** is pronounced 'C minus 2' – the **C** refers to the musical note, while the **'-2'** refers to the octave this C note is situated at. The '-2' octave is the lowest octave available in the MPC Software, and C-2 is the lowest note we can work with in the MPC environment (as we'll see later, it's actually ridiculously low, and effectively unusable for the majority of instruments)

As we move along the pads (moving in order from pad 1, pad 2, pad 3 etc), we can see the keyboard in the MPC Software is moving up a semitone at a time. Now, even though the actual note that pad represents is not displayed, we can easily calculate that as we know the chromatic scale. So in **Bank A,** if **pad 1** is the '**C -2'** note, **pad 2** must be the '**C# -2'**, **pad 3** must be the '**D -2'** note and so on.



When we get to **pad 13**, we've completed a full octave from the original C note, so we add '1' to the previous '-2' octave to get '**C -1'**.

Now select **bank B** and keep on pressing those pads in order until you reach **pad 9** and see the note **C0** selected. Hence **pad 9** in **bank B** represents the **C0** note in your keygroup program. This goes on until bank H where **pad 9** represents **C8**. Referencing the chromatic scale, we can work out that the final pad 16 is assigned a G8, which is the highest musical note we can use in an

MPC (and this one is ridiculously high, so you'll probably never actual use it in an instrument program)

Using Multisamples in a Keygroup program

Let's assume for a moment that you've set up a microphone in front of a piano and recorded every single note from it. All you'd need to do now is assign the correct sample to the correct pad in your MPC keygroup program. So your C1 piano sample simply gets assigned to the C1 pad in your program. Then your C#1 sample is assigned to the C#1 pad in your keygroup program. And so on, until all your pads are filled with samples that match the correct musical note that pad should have. The process of recording the individual notes from an instrument is referred to as **multisampling**.

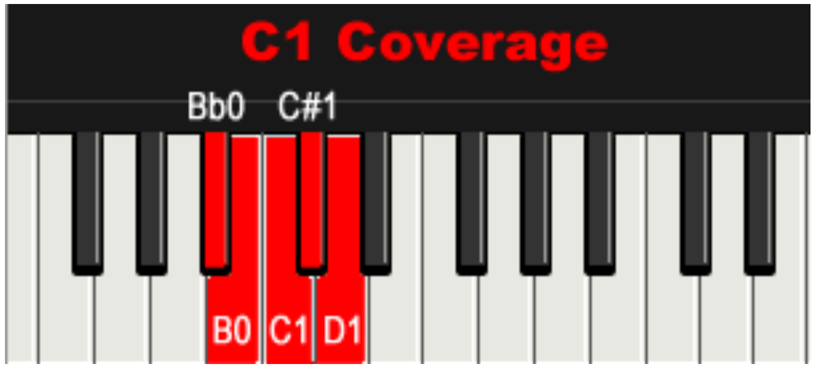
When we multisample we can record every single note from an instrument (often at multiple dynamic levels, but that's a more advanced tutorial!). However it is not always necessary to sample every note and instead we might only record a *selection* of samples from your instrument and then set up our keygroup program to fill in the gaps by 'tuning' the sample up or down to emulate the missing notes. We've previously tuned samples in a DRUM program using the '**SEMI**' parameter (found in the second SAMPLES screen of PROG EDIT), but keygroup programs are designed to handle all the tuning for us; but only if we set everything up correctly in the first place.

So, enough talk, let's build a very basic keygroup program to better understand the core features available to us.

Fundamentals of 'key ranges'

Let's consider a scenario where we only record three notes from each and every octave; **C, E, G**. We can assign these notes to their correct pads, and set up the keygroup program to fill the gaps using tuning emulation.

So in **bank C**, the **C1** sample is set up to not only cover the dedicated **C1** pad, it is also assigned to cover the **C#1** and **D** pads after it. Going the other way, that same **C** note can be set up to cover the **B** and **Bb**:



The keygroup program will handle this by automatically tuning the C1 sample up or down accordingly. As it's only tuning by a couple of semitones either way, there is no noticeable degradation in quality of sound. We can do something similar for the **E1** sample:

And the **G1** sample:

And so on over all the octaves naturally covered by the piano. This **'group'** of notes is called a **'keygroup'** and the range they cover is called a '**key range**'. Using this method, we can still make a usable multisample program using a quarter of the sounds and a lot less effort.

Sourcing & Configuring Your Samples

The most common way to source multisamples is to record them yourself directly from an instrument. If you are multisampling an acoustic or electric instrument (such as a piano or guitar) you would normally do this via the SAMPLE REC screen (we looked at in chapter 8) or via an audio track (we'll learn about audio tracks in chapter 14) - refer to the **Appendix** for more details on how to set up a microphone with your MPC.

Regardless of recording technique, one of the most important tasks you'll need to do with any set of multisampled sounds is to 'configure' the sounds correctly before attempting to build a keygroup program from them.

Naming Your Sounds

Firstly your sounds should be named logically. There's no set rules to how you name your samples, but the point is you need to use a system that helps you identify the instrument or sound patch that the sample was recorded from as well as its musical pitch.

So for example, if you recorded a 'C1' note from a piano this could be called **piano-C1**. Or you could go further and specify the type of piano, such as 'Grand Piano', or 'Steinway Model D'. If you were recording a patch from a synth you could include both the synth model name itself and the specific patch name, e.g. '**Massive-Wobbly Bass E2**'.

However, at this point let's keep things as simple as possible so let's just use the simple '**piano-C1**' format.

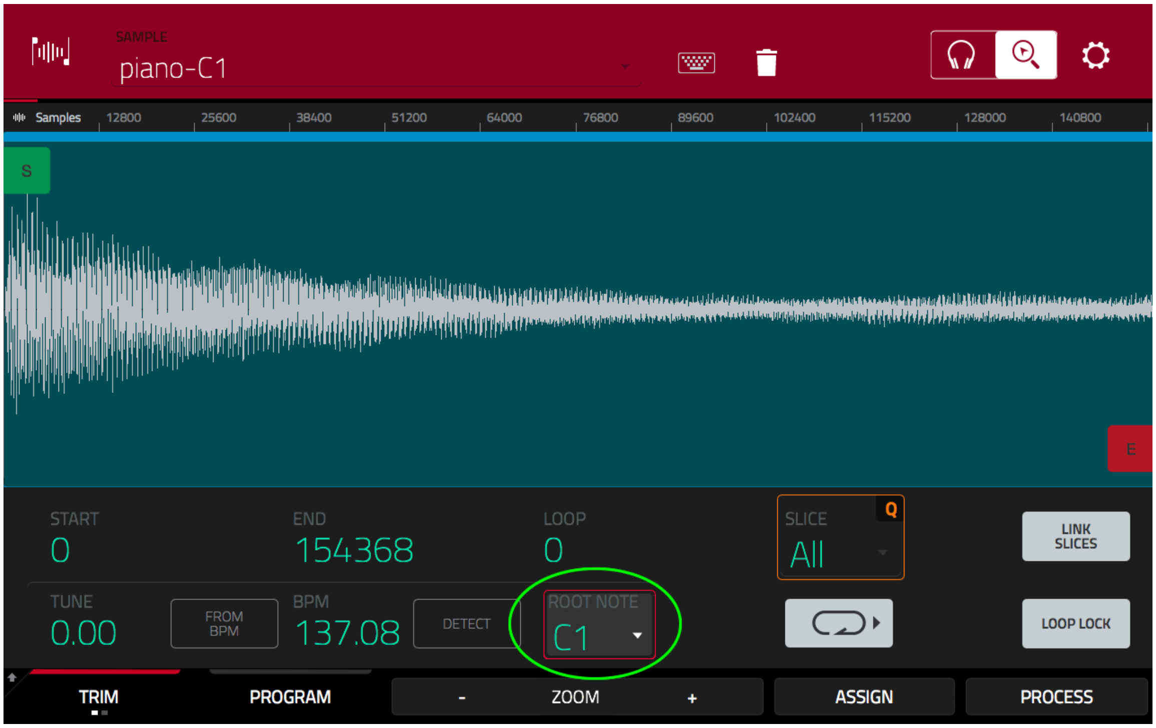
Configuring The Root Note

Logical naming of a sample is there purely for your own benefit so you can visually identify the source and pitch of a sample. The MPC on the other hand won't recognise the pitch of the sample from its file name, so we have to turn to '**Root Notes**'.

Root notes tell samplers what musical pitch (or 'key') a specific sample was recorded at. A root note can be configured in two ways; at 'sample level' and/or at 'program level' - you can use either, but let's initially look at setting the root note at sample level (we'll look at setting it at program level in chapter 26).

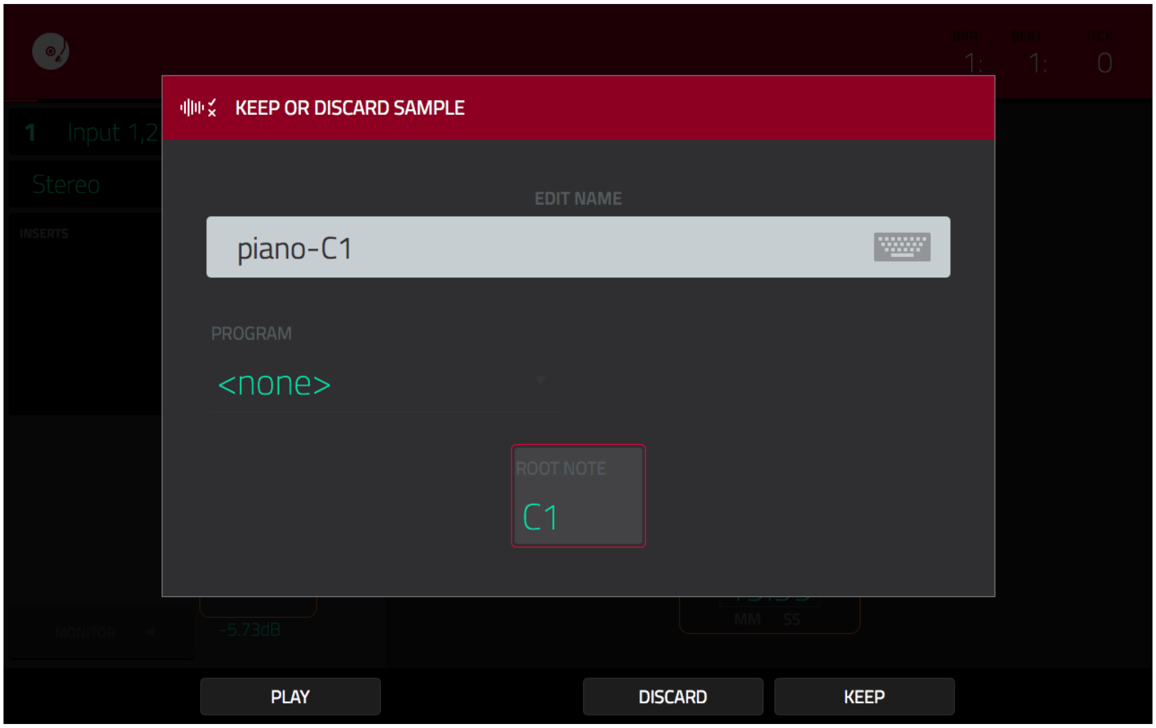
Go to the **BROWSER** and in the **chapter 10** folder single tap the folder '**Piano Sounds**'; hold down **SHIFT** and select **LOAD ALL**. This will load a collection of piano multisamples that I recorded previously.

Go to **SAMPLE EDIT > TRIM;** the '**ROOT NOTE**' parameter is already set to '**C1**' for the 'C1' piano sample:



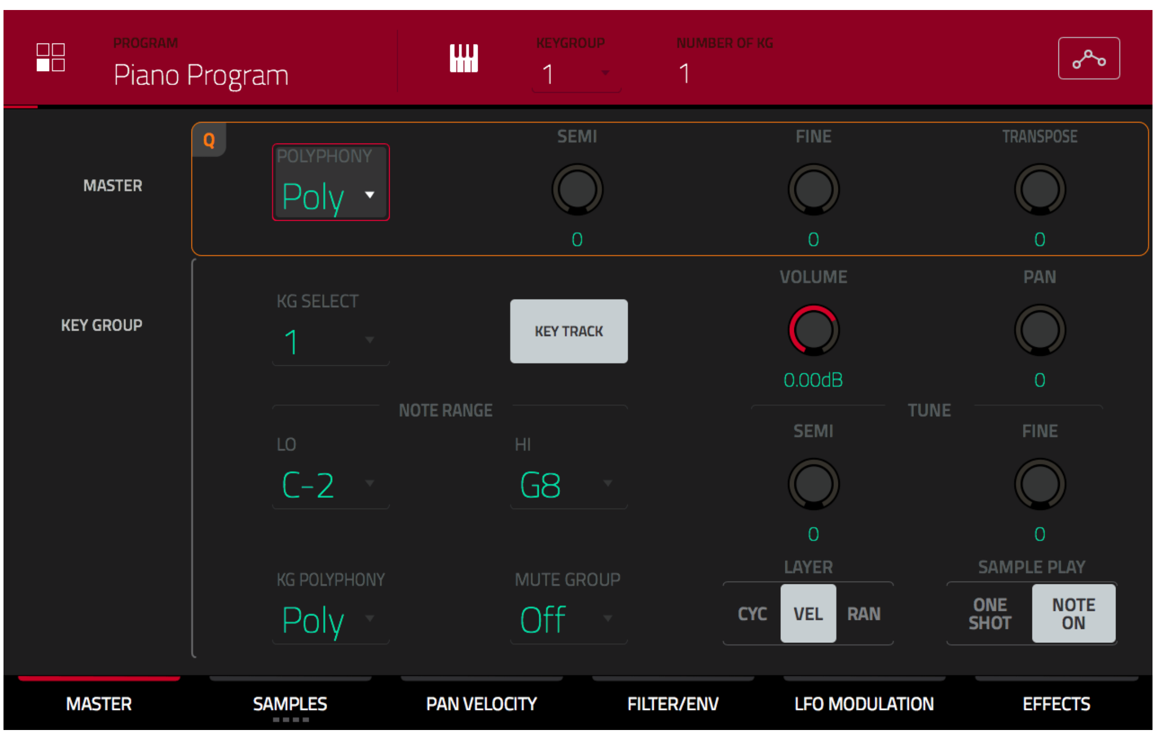
If a sample has never had a root note set you'll find the root note defaults to C3 - in which case just tap the '**ROOT NOTE**' value and set it to the correct value with your data wheel.

You can also set the root note during the initial recording process (assuming you use the SAMPLE REC screen for recording). After recording your sample, you'll have an opportunity to set the root note in the '**Keep or Discard Sample**' screen:



Setting Up Your First Keygroup

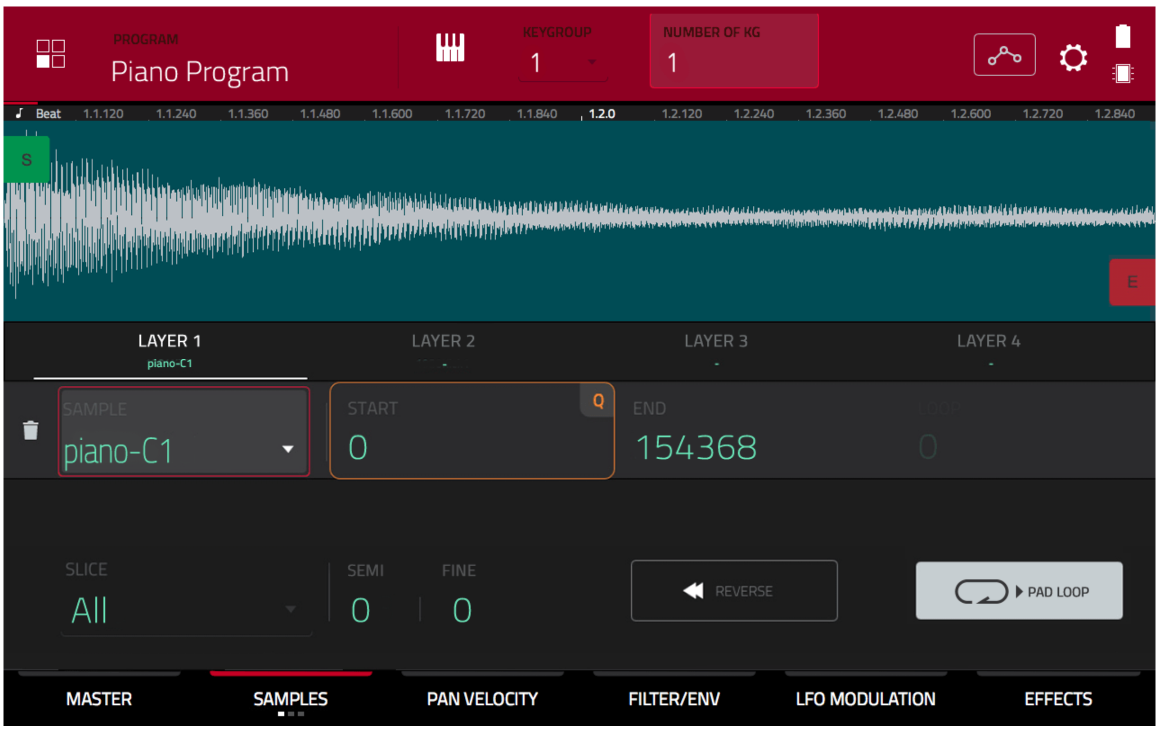
In **PROGRAM EDIT**, go to the **MASTER** screen:



When you first create a KEYGROUP program, your MPC assigns that program one single **‘keygroup’** which spans all 128 pads in the MPC - the number of keygroups in a program is shown at the top of the screen (‘**NUMBER OF KG**’). As you can see, there is only ‘**1**’ keygroup in this program.

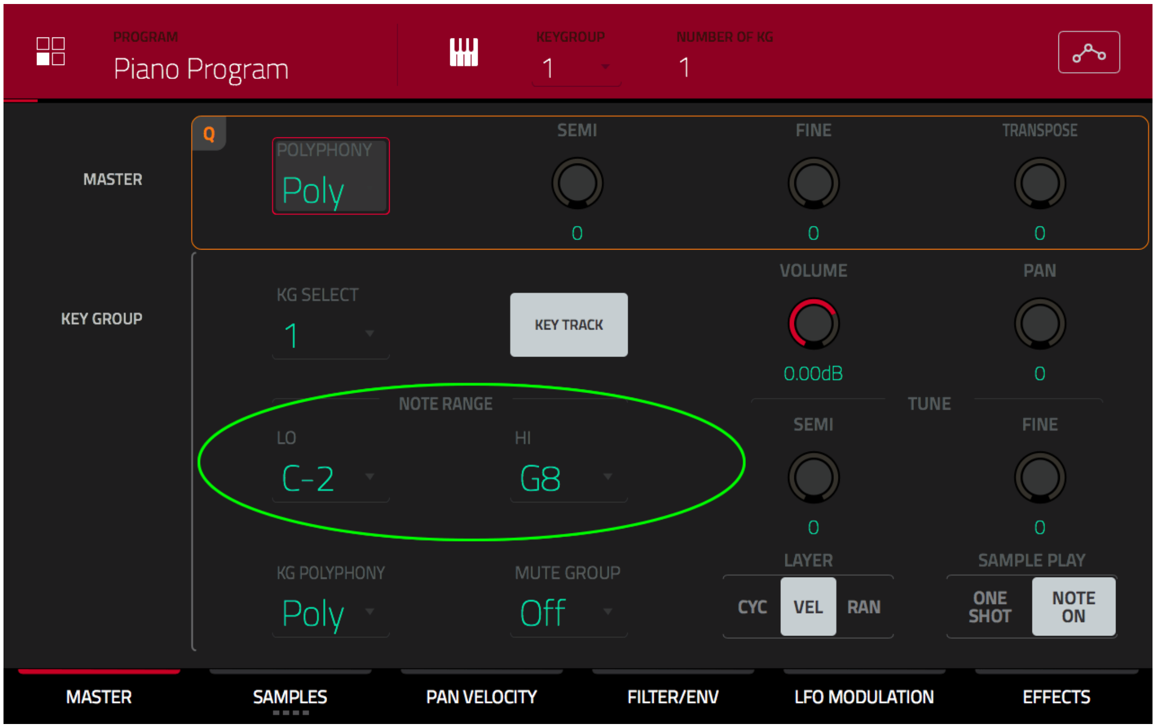
To assign a sample to this keygroup, go to the **SAMPLES** screen, which is identical to the DRUM program version. However this time you are not going to assign this sample to a single ‘pad’, you are assigning it to an entire **keygroup**.

You can assign the **piano-C1** sample to **layer 1** in our keygroup using any of the methods you’d use in a DRUM program.



With the piano-C1 sample assigned to your keygroup, start playing your pads. I suggest first selecting **BANK C** as the *usable* range starts here. You should hear that your piano sample has been automatically ‘spread’ across all pads and has been ‘tuned’ to run chromatically. If you try out some other pad banks, you’ll see that your one sample has in fact been mapped across all 128 pads in the MPC, and you have a very wide range of tuning (as I mentioned previously, the extremes in banks A, G and H are in most cases beyond the natural ‘range’ of any acoustic instrument).

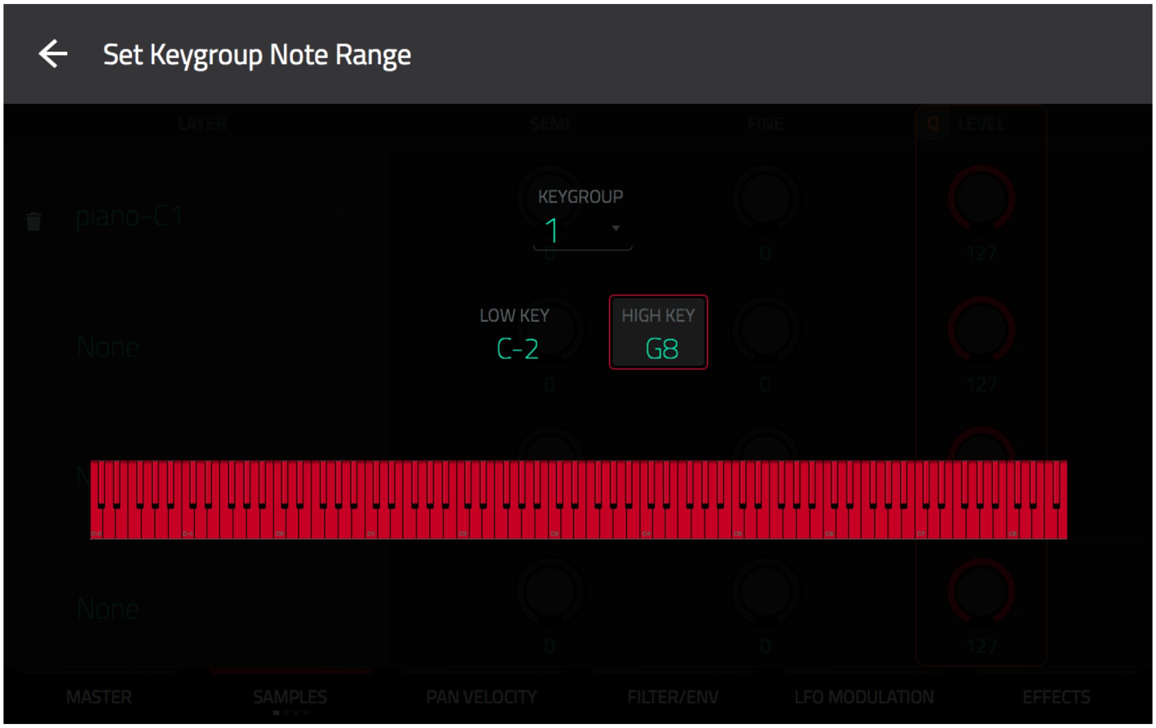
We can view and edit the range covered by this keygroup in the **MASTER** screen. Go to **PROGRAM EDIT > MASTER**:



You can see via the **KEYGROUP** parameter in the top of the screen that the current keygroup is **1**. Then on the third row we have the '**NOTE RANGE**' for this keygroup, which is set to the maximum range of '**LO: C-2**' and "**HI: G8**'. This represents all 128 available pads in the MPC program.

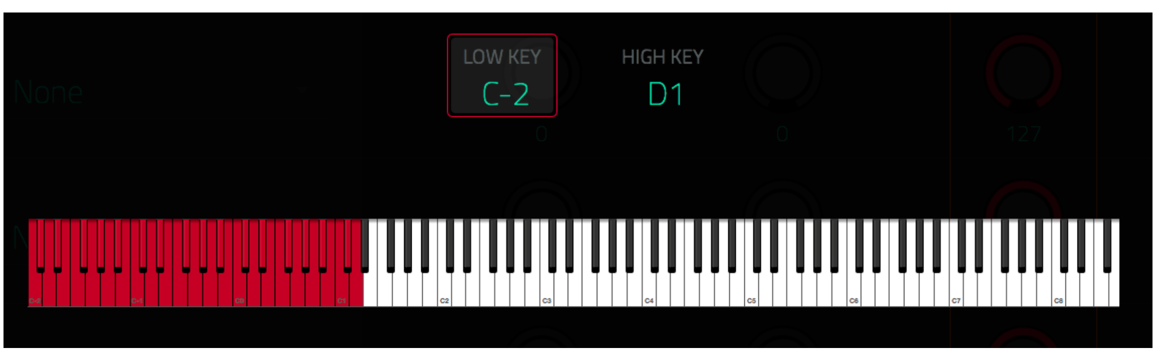
You can access a more 'visual' representation of our keygroup by clicking on the keyboard icon at the top of the screen:

This brings up the following **Keygroup Note Range** screen:



As you can see, every key on the keyboard is red indicating that the currently selected keygroup is spanning your sample across **all** available notes, **C-2 to G8**.

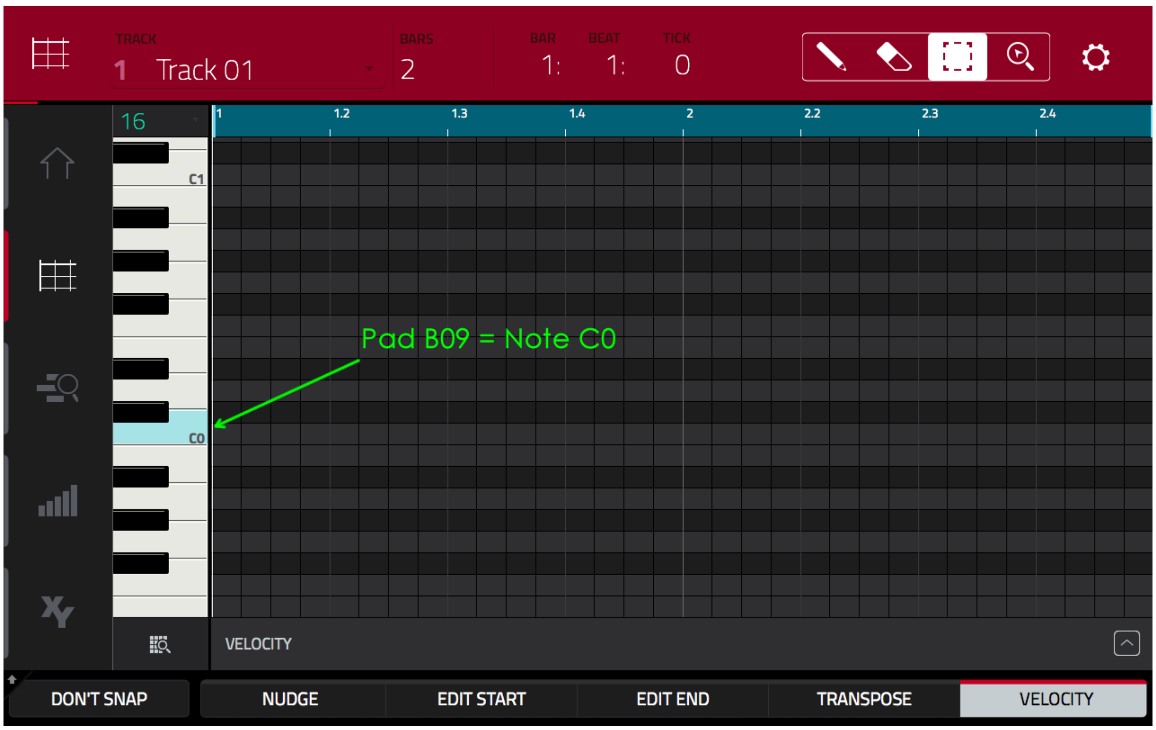
We'll need to limit the range covered by our piano-C1 sample as currently most of the sounds crated across the key range sound terrible due to the excessive tuning. So let's limit this C1 sample to go no higher than the **D1** note (two semitones). To limit the range of notes that our sample is mapped across we simply adjust the note range values in the **MASTER** screen. So set '**HI**' to read **D1**.



With pad bank C still selected, start playing the pads starting at *pad* **C01** (i.e. pad **01** in bank **C**) – when you reach pad **C08** and beyond all the pads are silent – it's the same for banks D to H.

So we've successfully limited the range of notes our sample is mapped out across; **C-2 to D1**. Now obviously it would be nice to limit the 'low note' value so our piano-C1 note isn't relied on entirely for all our low notes (maybe take it down to B0. The problem is that piano-C1 is the lowest sample we have recorded, so that decision is made for us.

However you could set a low note value regardless, because remember, this is a piano and it has to sound real, so we should put a limit on how much this poor C1 sample is going to be tuned down. Select **BANK B** to access **pad B16** and begin playing the pads until you hit the point where you realise that this piano is just too far tuned down. I reckon the last pad worth using is **pad B09**, which if you look at the virtual keyboard in GRID VIEW is showing as a **C0**.



Adding Keygroups

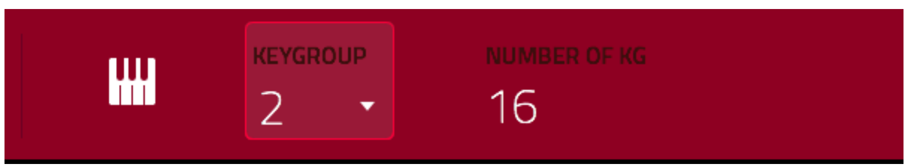
With our first keygroup configured, it's time to start introducing more multisamples into our instrument program. Now for each octave here we've got C, E and G, so the next sample from C1 will be **E1**, followed by **G1**, then **C2**. Each of these samples will cover their own mini 'key range', just like C1 does. To add each of these samples to the keygroup program we'll need to create a unique keygroup for each of them and then assign each keygroup an appropriate range, just like we did with our first keygroup.

So as we saw previously in this chapter, our **C1** sample covers up to **D1**. Then the **E1** sample covers **Eb1** to **F1**, our **G1** sample covers **F#1** to **A1,** and finally **C2** covers **Bb1** to **D2**.

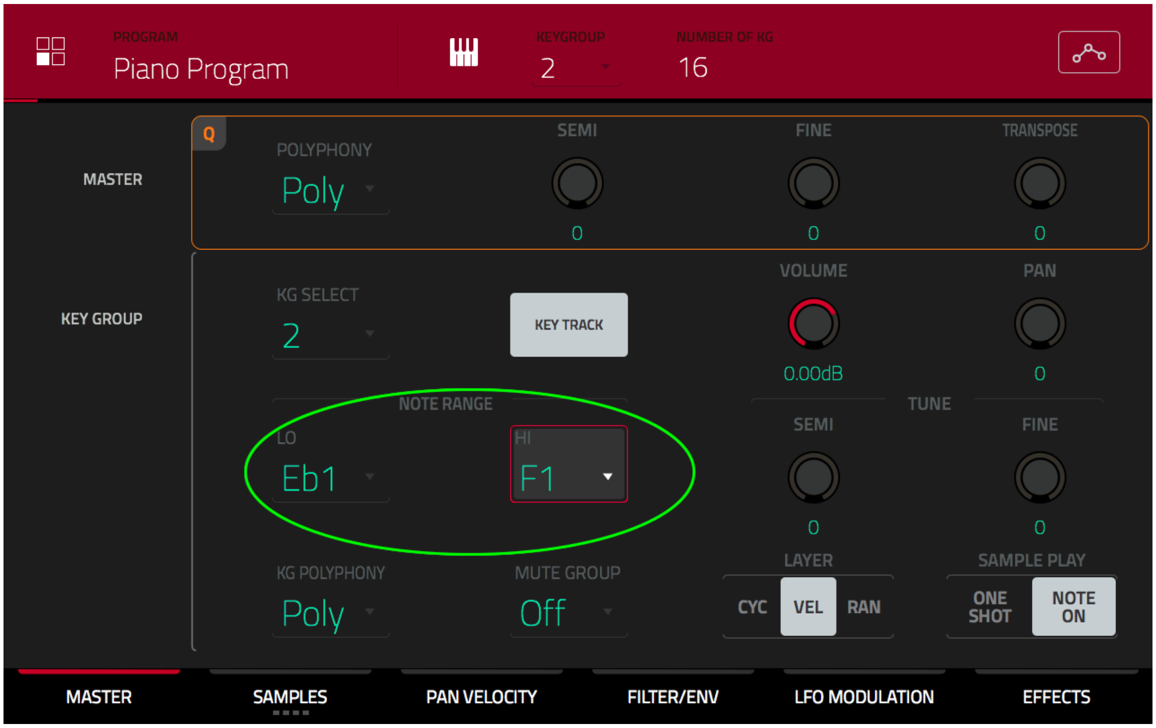
You can create up to 128 unique keygroups, where each one can have a note range covering anything from a single note right up to the entire 128 note span. As far as I am aware you can happily create more keygroups than you need because unless you've added an actual sample to a specific keygroup it's effectively non-existent as far as the MPC is concerned.

To add additional keygroups to your current program, tap the **NUMBER OF KG** parameter in the **PROGRAM EDIT** > **MASTER** screen:

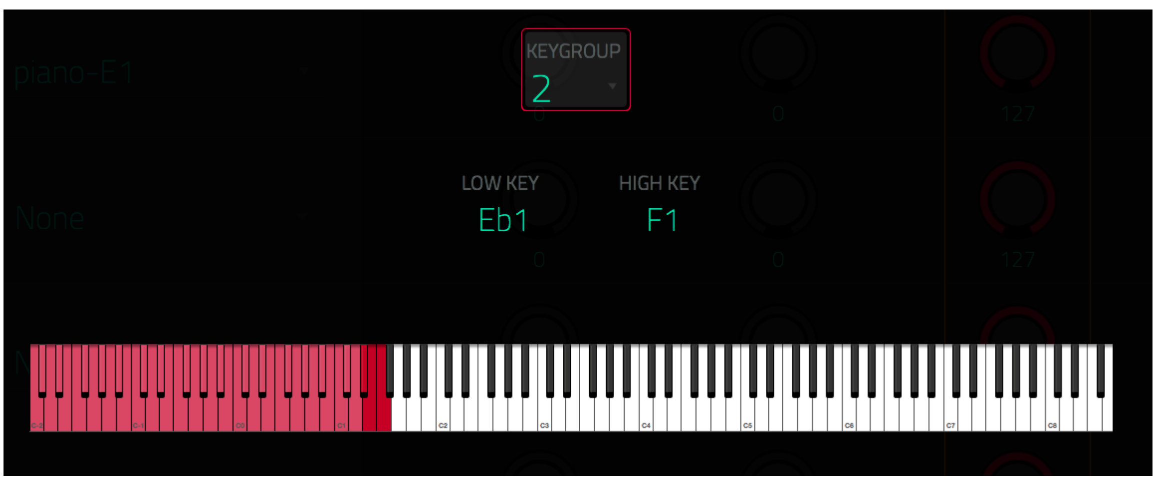
Turn the data wheel until you have a value of **16** (i.e. 16 keygroups). So set **16** keygroups and hit **DO IT**. Now tap **KEYGROUP** and select keygroup **2**:



By default, a new keygroup is set to the entire note range of C-2 to G8, so we're now going to change this to the range we indicated earlier; **Eb1** to **F1**, so set **'LO'** to **Eb1** and **'HI'** to **F1**.

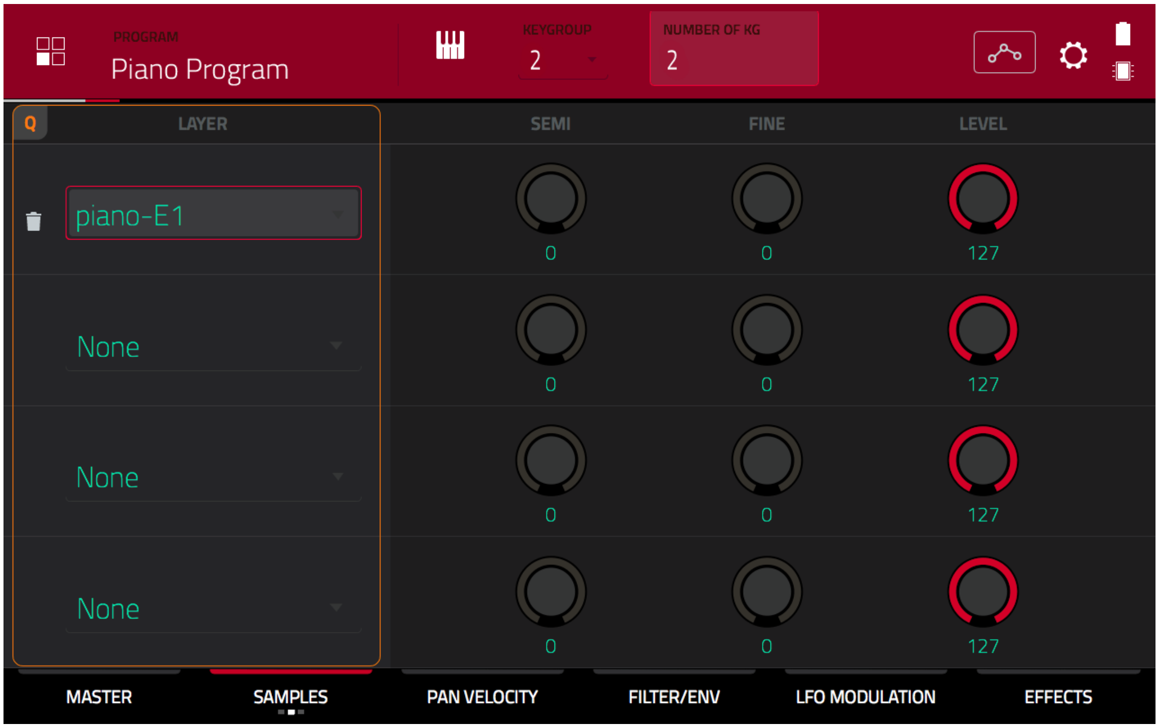


Check the **Keygroup Note Range** screen now:



Our current keygroup is in red, while notes that are currently assigned to other keygroups (i.e. so far, keygroup 1) are in pink.

Go to **SAMPLES** and assign the **piano-E1** sample to **layer 1**. Just like a DRUM program, you have the option to also assign samples in the '**second**' **SAMPLES** screen:



Now go to **bank C** and play pads **1** to **10** - you should hear a nice chromatic progression.

We can now do the same for all our samples - select **keygroup 3**, assign the **piano-G1** to **layer 1,** and set the range to cover notes **F#1 – A1**. **Keygroup 4** takes sample **piano-C2** and covers the range **A#1 to D2**. **Keygroup 5** takes **piano-E2** over the range **D#2** - **F2**. And so on for all samples and keygroups. The pattern is as follows (assuming all samples are C, E and G from each octave):

C notes cover: **A# to D** (e.g. A#3 to D4 – note that the D is in the next octave)
E notes cover: **D# to F** (e.g. D#3 to F3)
G notes cover: **F# to A** (e.g. F#3 to A3)

For your last sample, you can either set the **'high Note'** to be **G8**, but again if you want to limit its useful range, you can do that.

You might find it quicker to first assign all the correct samples to all the keygroups (in the **SAMPLES** screen), and *then* go back to the **MASTER** screen and assign the note ranges (rather than going back and forth between Samples and Master for each keygroup). You can of course assign ranges to your exact taste - some samples can be tuned up or down more than others, it's up to you – for example while a **G** note is currently covering **F# to A**, you could easily change this to cover **F to G#** - you would have to then adjust the ranges covered by the C and E samples either side of it.

Congratulations, you've just created your first keygroup-based multisample program! Remember, bank A contains incredibly low notes, so select bank C to hear where the more realistic note range begins.

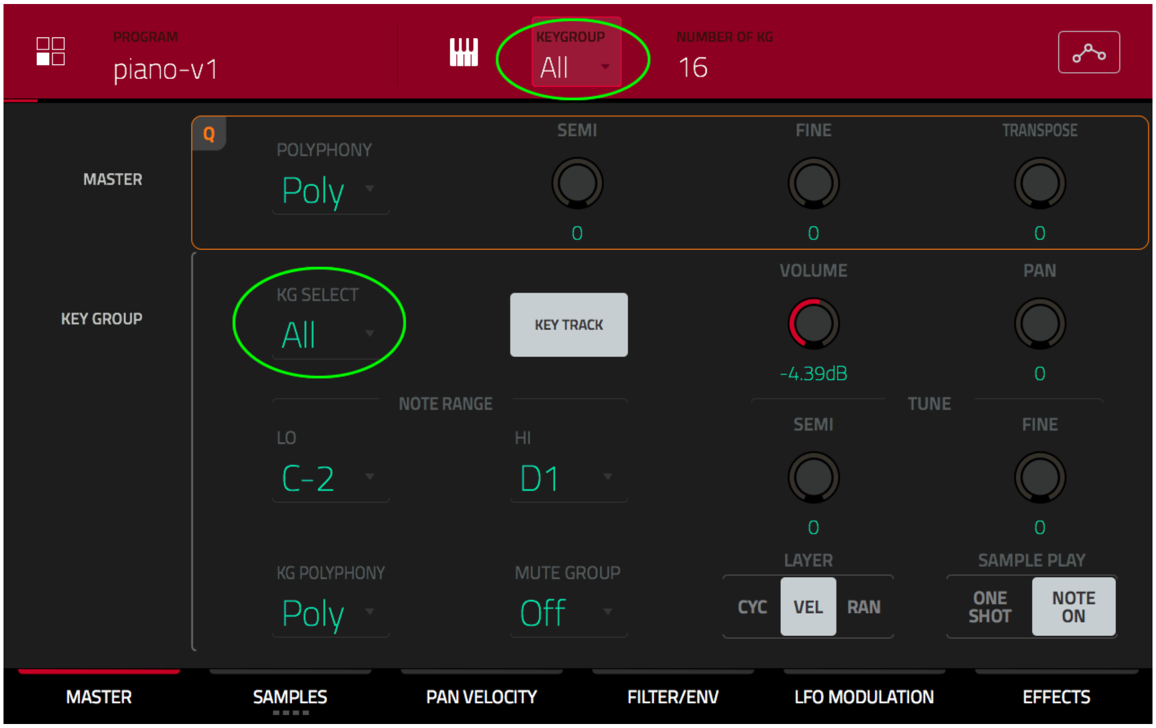
Applying ADSR

You may not have noticed but, unlike DRUM programs, the default keygroup program '**SAMPLE PLAY'** setting is **NOTE ON**, not ONE SHOT – after all, this is a program type primarily designed for instruments, so it would make sense for the sample to play for the duration we hold a pad for.

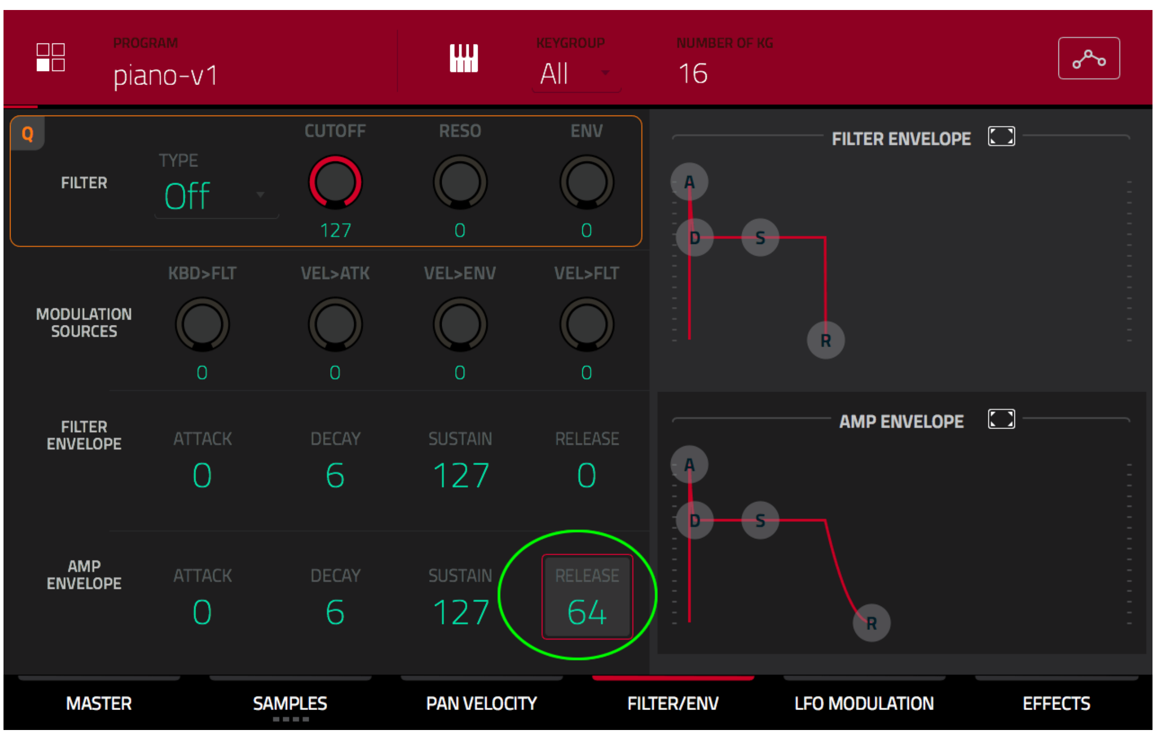
But just like our bass sample in the 16 LEVELS tutorial we can clearly hear some nasty clicks at the end of these samples whenever we release the pads, so it's time to head over to the **AMP ENVELOPE** settings to add some gentle fade to the end of each sample using the **release** parameter.

Nearly all program parameters in a keygroup program are applied individually to each specific *keygroup*, rather than to individual pads, so you could set a different release value for each individual keygroup if you wished, which is often appropriate, as lower range piano notes tend to have a longer release compared to higher range piano notes.

Initially though, let's set the same **release** value to all keygroups equally. To do this we simply select all the keygroups together – in the **MASTER** screen, tap the KEYGROUP parameter at the top of the screen, or tap the '**KG SELECT'** parameter and turn the data wheel anticlockwise until it reads **'All'** (or double tap and select **All**).



With all keygroups now selected, head over to **FILTER/ENV** where we have the standard **ADSR** amplitude envelope for Note On settings. Set yourself a nice **release** value – **64** works well.



Remember just because ‘All’ keygroups are selected, you can of course still play each individual pad to preview the changes you are making, so this gives you a chance to set a nice release that works well for most keygroups.

If you select some of the keygroups individually (change **Kg Select** from ALL to a particular keygroup number) you can see they all now have a release of 64 applied. You can now of course tweak the release setting for any keygroups where you feel they may need slightly longer or shorter release. For example, I selected **keygroup 1** and added a **release** of **75** for these much lower notes, and then selected **keygroup 2** and added a **release** of **70**.

Load up **Piano Program.XPM** for my version.

011 Creating Instrument Melodies & Chords

From the **chapter 011** folder, load up the project file **Demo Beat.xpj**. This project loads the drums and bass line we've been working on previously to sequence 1 (**16 Levels**). It also loads the **Piano Program** keygroup program that we built in the previous tutorial. Let's use this piano program to add a piano rhythm track to our sequence.

MIDI Sound Sources

As we discovered in the previous tutorial a keygroup program is laid out chromatically like a keyboard on a MIDI synthesiser, and each pad in the program produces a MIDI note at a specific musical pitch.

This makes keygroup programs perfect for playing musical instrument performances. However the MPC is not only limited to playing back sample based MIDI instruments; we can also use our sequencer to play back sounds from attached 'hardware' MIDI synths, keyboards and sound modules – the MPC X can even control analog synths. And in 'controller mode' we can also play 'software instruments' loaded from VST, AU and AAX plugins or even the MPC's built in software instrument, Hybrid 3.

All these instruments are just different examples of MIDI sound sources. Fundamentally all these instruments are 'recorded' identically by the MPC because the MPC does nothing but record MIDI events; these events (or instructions) are then sent back to the MIDI sound source so it can play back the correct sounds and notes.

In fact once we've recorded a MIDI track using one type of sound source, we can use that same MIDI event data to trigger sounds from any other type of MIDI sound source simply by changing which sound source the MIDI track points to!

Now, if you are in 'standalone mode', and have no external MIDI gear, then your options are initially limited to using keygroup programs, so for maximum compatibility I'm initially going to record everything with the Piano keygroup program we built previously. However later in this chapter we'll see that all the MIDI tracks you record with a keygroup program can be easily 'switched out' with any other MIDI sound source at a later date.

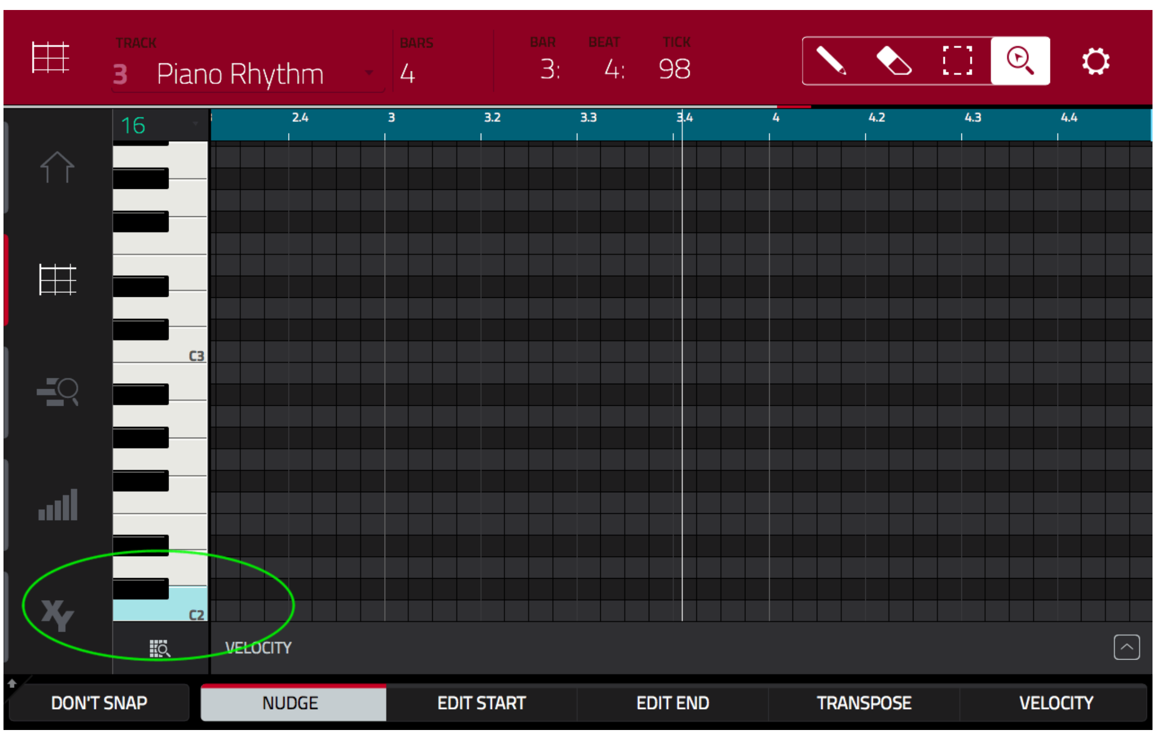
Finding the Key

First things, first, we need to set up a track to record our initial piano parts. In the previous chapter we'd added our KEYGROUP program (**Piano Program**) to track 3 of the **16 Levels** sequence. Rename this track **Piano Rhythm**.

Select **BANK E** and play your pads to hear your piano program play.

A common obstacle when dealing with a mix of sampled sounds and MIDI instruments is matching musical key. In our current project we wrote a bass line created from a 'random' bass pulled from a sampled song, so off hand we don't know what key this note was originally in and hence we don't know what musical key our bass line is in.

Press **PLAY START** so our beat plays in the background. Now start playing your piano program to see if you can 'match' the notes played in your bass line. Select **BANK D** and play pad **D01** – you should hear that this note matches the initial bass notes in bar 1. Head over to **GRID EDIT**, hit pad **D01** and take a look at the on screen keyboard – you should see that the bass note is a **C2** – it's actually a lower octave than this but that doesn't actually matter, it's the '**C**' that's important.



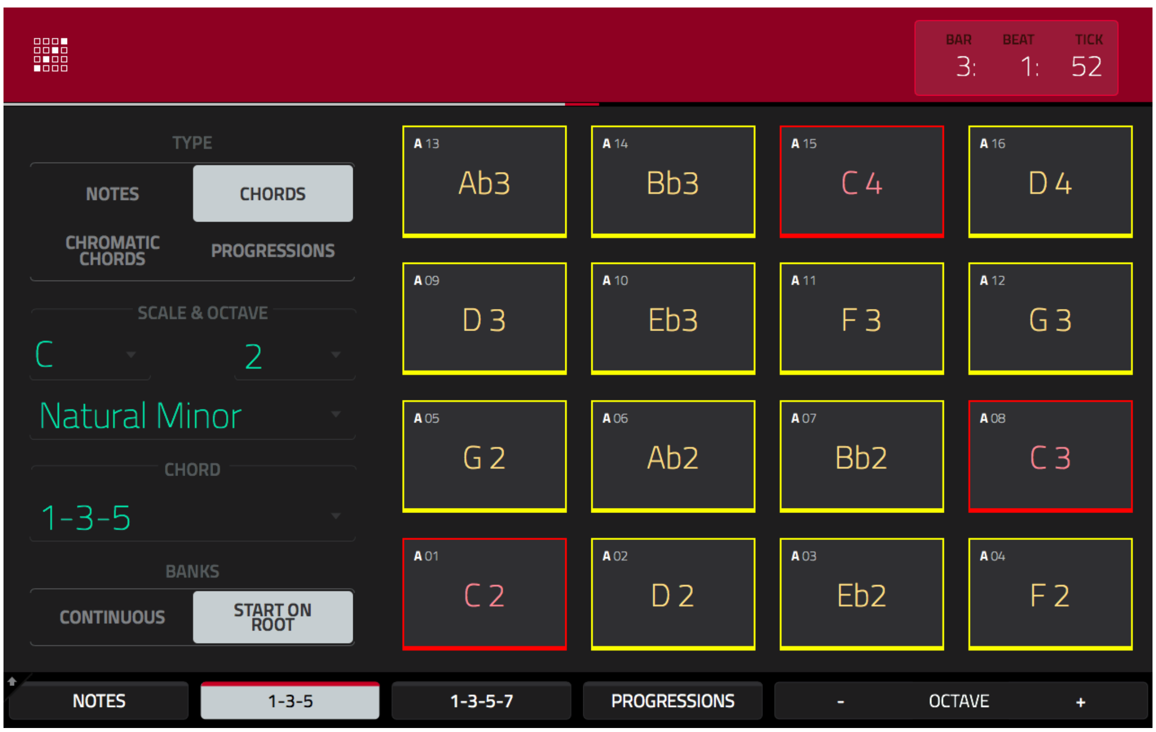
The note played at the start of bar 2 is the same key as pad **D08**, which is a '**G**' note.

So basically our bass line seems to be indicating our song is in the key of '**C**' (it could be major or minor) with the chords going from a C to a G every 2 bars. This is a nice and simple progression to work with.

Recording a Piano Rhythm Track

Now, if you don't have any experience with musical scales and chords, creating an instrument performance can be a bit of a hit and miss affair. If you are comfortable playing a MIDI keyboard, you can connect one of these to your MPC and play your keygroup instruments this way (see **Appendix A** on how to set up MIDI equipment).

However if you prefer to use the pads, the MPC has a dedicated mode that can help you; **Pad Perform Mode**. Go to **MENU > PAD PERFORM**:



The first parameter we see is '**TYPE**'. Set this to **CHORDS**. A *chord* is a collection of notes from a scale that are played together in unison, typically used to create an underlying rhythm part for a song. Press the **1-3-5** button, select **BANK A** and a **C 'Natural Minor'** scale with an **OCTAVE** of **2**.

Play pad **A01** and you'll hear that it now plays back a complete chord rather than a single note. As we selected '**1-3-5**', the MPC is playing a chord consisting of the 1st, 3rd and 5th note in the selected scale, which for C Minor would be C, Eb, G . Remember how we decided our bass line was in the key of C? Well that 'C' chord on pad A01 definitely sounds pretty good over the first part of the bass line.

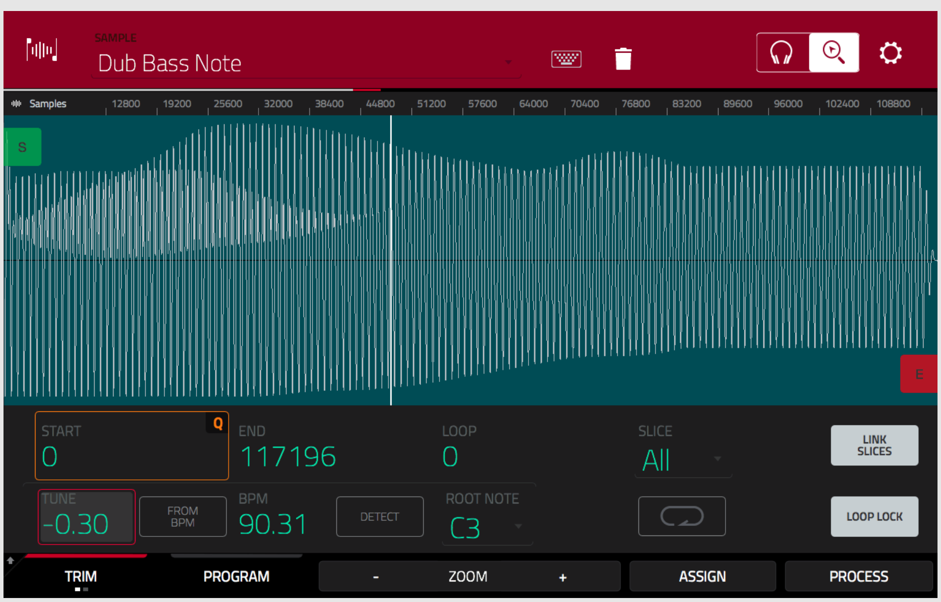
When you hit pad **A02**, the MPC will just move that 1-3-5 pattern up one note in the C Minor scale, so it's now playing a chord starting on D, along with F and Ab (this is actually a D diminished chord). The next chord up on **A03** is Eb-G-Bb, which is an Eb major chord. Basically any pad you press when in 1-3-5 mode will play a chord compatible with the C Minor scale (or whichever scale you select under 'Scale').

If you now press **1-3-5-7** the MPC performs a similar operation, but this time it will add an additional 7th note, which will turn our major and minor chords into major 7ths and minor 7ths. These tend to sound much more jazzy.

Hold down **SHIFT** to access more scale chords; **1-4-5** and **1-3-5-7b** (flattened 7th).

Tuning Samples to Concert Pitch

One thing I immediately heard was the bass line sounds a little 'sharp' compared to the piano which would suggest the bass wasn't recorded quite at concert pitch. To resolve this, head over to **SAMPLE EDIT**, select the **Dub Bass Note** sample and take the **TUNE** parameter down a little – remember to hold down **SHIFT** while you turn the data wheel.



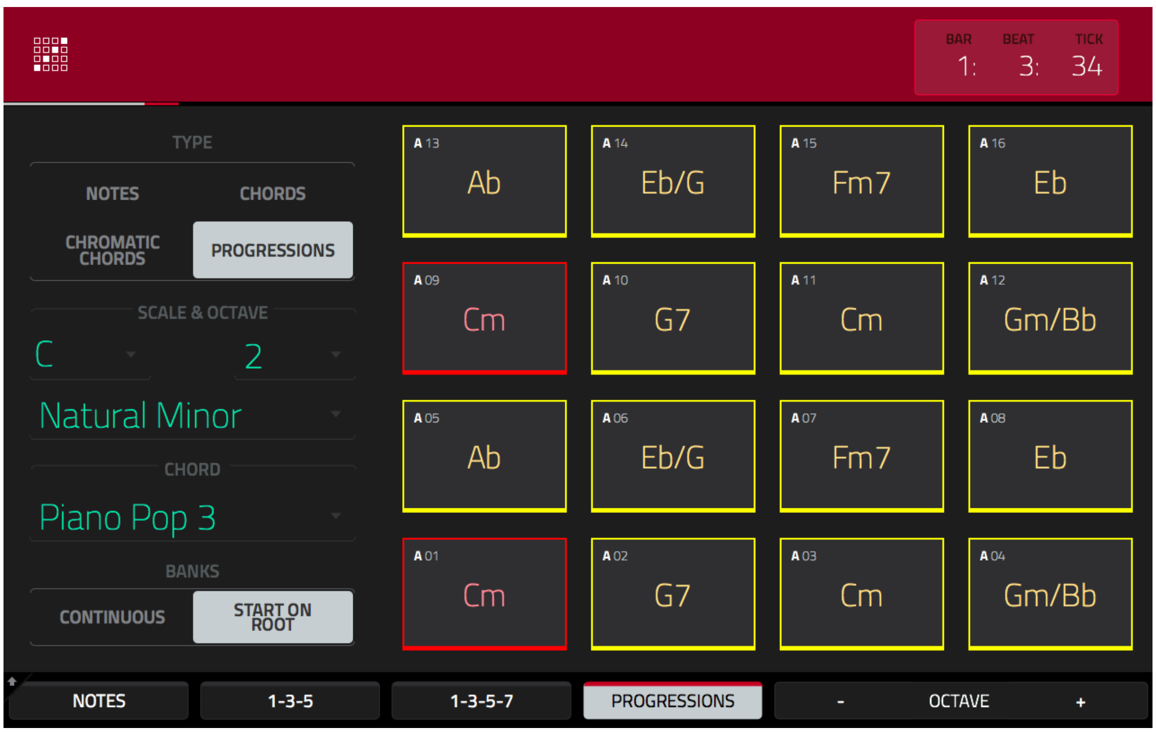
Make sure your beat is playing in the background so you can hear how the changes in tuning affect the way the bass line sounds compared to your piano parts. I found a TUNE Of -0.30 was pretty much perfect.

Remember not everyone has 'perfect pitch' so it's more common to not be able to hear these subtle differences.

Chord Progressions

PAD PERFORM mode also features dedicated chord 'progressions' which provide more structured chord combinations that feature the types of changes you'll often hear in more complex song arrangements, with changing root notes and chord inversions.

Tap on the **PROGRESSIONS** button. Under **CHORD** you'll now find a number of chord progression presets based on musical genre. Select **Piano Pop 3**:



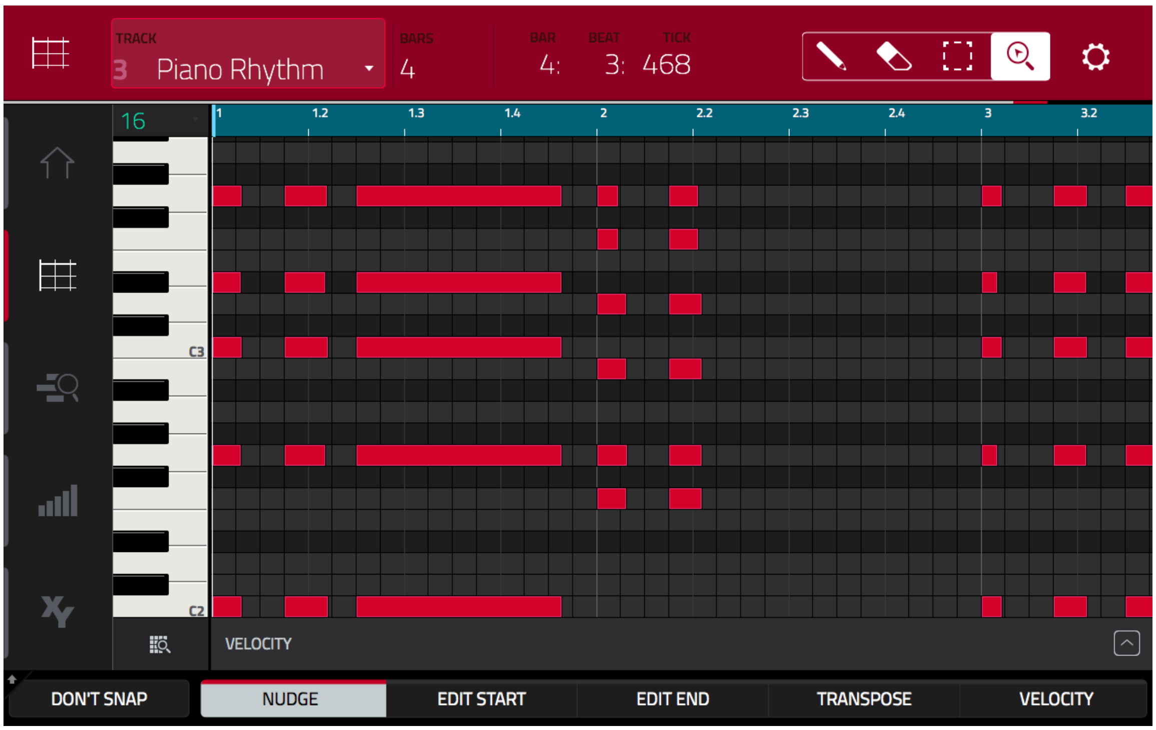
We've now got a whole bunch of chords that work well in the key of C minor. Start playing your pads and you'll hear just how easy it is to create a professional sounding jazz chord progression.

Now, remember how we worked out that the second (and fourth) bar of the beat used a 'G' instead of a C? Try the following; at the start of bar 1 and bar 3, play pad A01 (C minor), and at the start of bars 2 and 4, play pad A02 (a G7 chord).

Sounds pretty good! Notice that pads A09 and A10 also feature Cm and G7 chords, just an octave higher than the ones on A01 and A02. For a fatter chord sound, try playing these together, so A01 with A09, and A02 with A10.

How you play them is up to you and the vibe you want to set. Short 'staccato' stabs are quite common in hip hop, while a single long chord at the start of each bar gives a more laid back RnB ballad feel.

Head over to **sequence 2** (**Piano Chords**) to hear my version on track 3 (**Piano Rhythm**). Notice that in **GRID VIEW**, my single pad hits are not just producing a single sequencer event; instead the MPC automatically adds the required events to build each complete chord for you:



Creating Custom Progressions

While Akai have provided an impressive range of ready-made progressions, you might prefer to build your own arsenal of selectable progressions from your own musical performances, and this is very easy to do, however these need to first be created in the MPC Software.

Fire up the MPC Software and load the **Demo Beat** project we've been working on from the **Chapter 11** folder. Select **sequence 14** - **Custom Progression** and hit PLAY START.

This is a 12 bar piano sequence, with the piano rhythm on track 1 and the piano sol on track 2. Let's convert the chords used on track 1 into a custom chord progression for Pad Perform mode.

With t**rack 1** of our short piano piece on **sequence 1** selected, click on the '**Tools > Convert Track to Progression**':

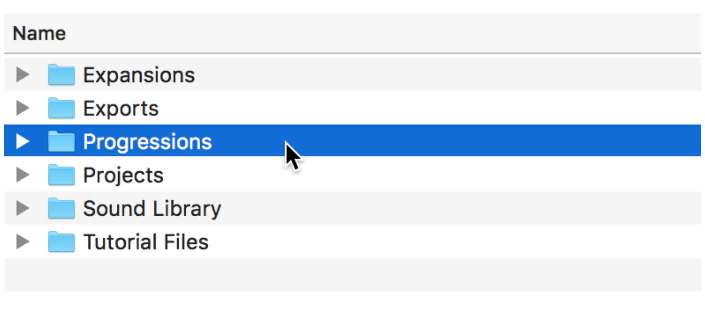
Here the MPC has analysed the events on the currently selected track, producing an accurate breakdown of the chord structure that was originally played. You are free to override and tweak anything that it might have got wrong, and once you are happy, you can give your progression a name (the default is to name it after the current sequence) - rename it to '**My Custom Progression**' and hit **Do It.**

The progression is now saved on your hard drive and is now available to select as a chord progression in **Pad Perform** mode:

To use this progression in standalone mode you'll need to copy the progression file to your standalone 'attached' MPC disk. Custom Progressions are found in the following location;

Mac: Library > Application Support > Akai > MPC > Progressions
PC: C:\Program Files\Akai Pro\MPC\Progressions

Connect your MPC disk to your computer, either directly or via **controller mode** (check out **Appendix B** for details on how to do this). Now copy the entire '**Progressions**' folder to the root location of your MPC disk:



Go back to standalone mode, select **PAD PERFORM** and your custom progressions are now available to use.

Methods For Adding Lead & Melody Lines

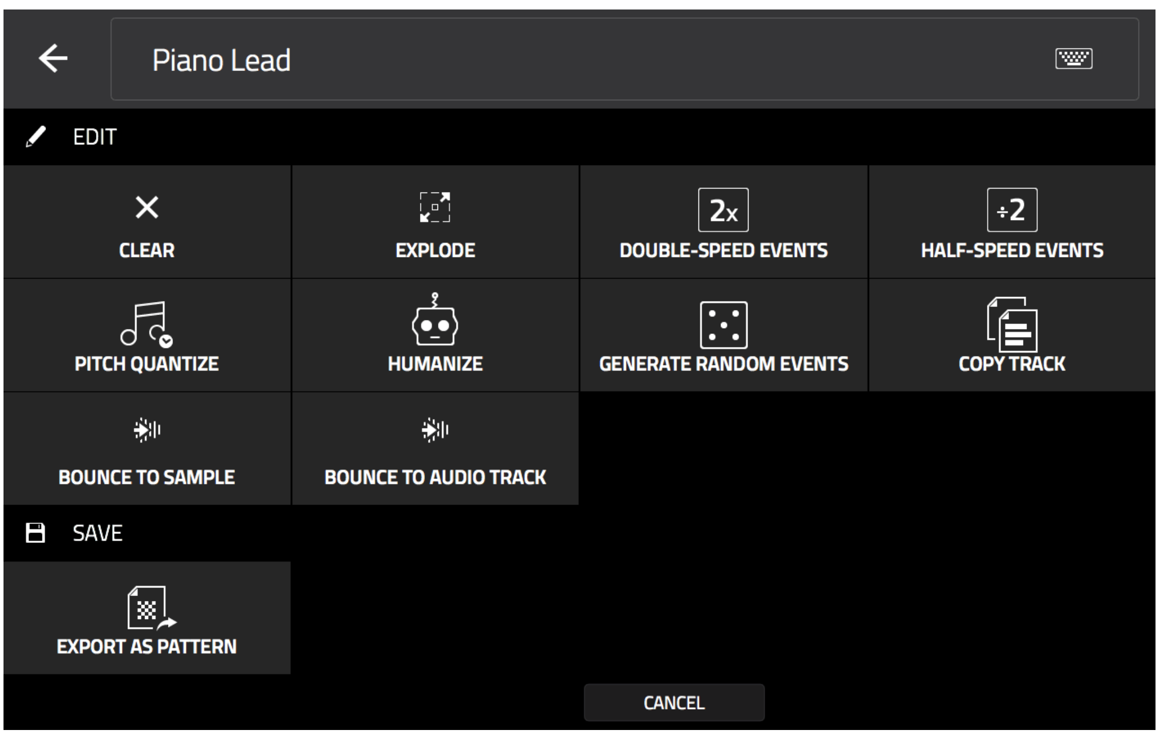
With a solid rhythm laid down it's time to think about adding some 'melody' to your beat, such as leads, riffs or solos. There are many ways you can do this in the MPC - the 'traditional' MPC method of recording MIDI-based leads and solos is of course to just play your instrument 'naturally' using your preferred controller, be it using the pads or an attached MIDI keyboard. If you have experience or training in piano playing then this is most likely to be your initial 'go to' option.

However even if you have great piano playing skills you should still consider the other options available to you as this will often lead to all sorts of interesting and creative experiments that would never have occurred to you in your more familiar and comfortable set up.

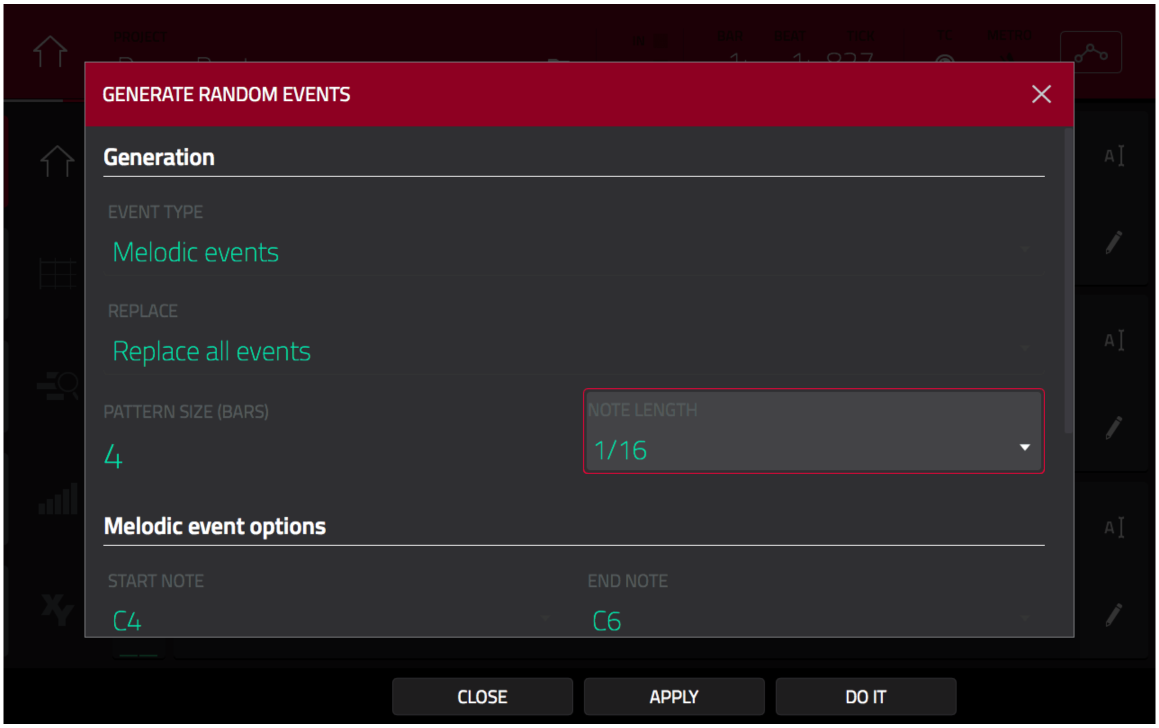
Generating Random Notes

Select **track 4** in the current sequence and add our **Piano Program** to this track. Call it **Piano Lead**. Alternatively select **sequence 3** (Piano Melody) where I have already created the Piano Lead track for you.

If you need a creative kick start, this feature might be perfect for you. Go to **MAIN** and on the **track** row select the **pencil** icon:



Tap on **GENERATE RANDOM EVENTS**:



This function simply lives up to its name as it will generate a bunch of random events on your track based on the criteria you set. That said, I'm not entirely sure that all the parameters on this page do exactly what the manual claims they should do, but regardless, this is a cool little feature that will certainly kick start some ideas for you.

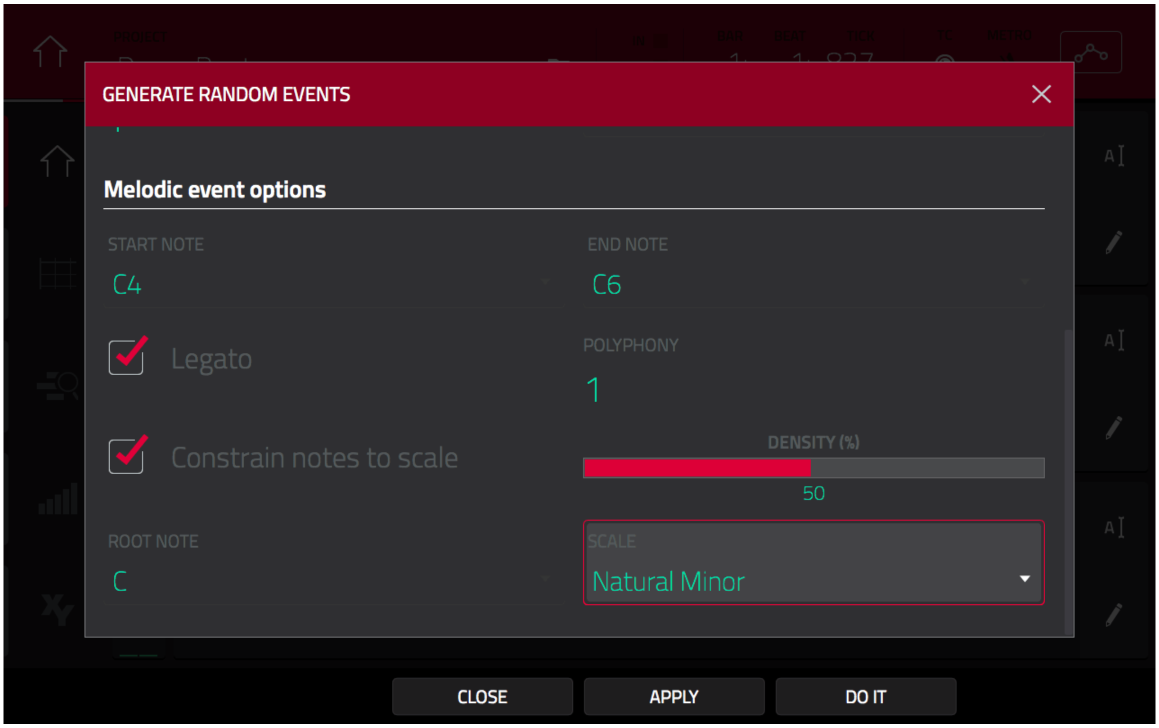
Let's start with some basic default settings. If the current track is a keygroup, MIDI or plugin track then the **EVENT TYPE** will be pre-selected for '**Melodic Events**', so leave this untouched.

Leave **REPLACE** set to '**Replace all events**' as in this example we'll be continually generating new melody lines until we find something that sounds good.

PATTERN SIZE is automatically set to cover our entire sequence (**4 BARS**) so let's leave this as it is.

Set **NOTE LENGTH** to **1/16** - we'll look at what happens when we adjust this later in this tutorial.

Drag your finger upwards to reveal the '**Melodic Event Options**' at the bottom of the page:



We'll use the **START NOTE** and **END NOTE** parameters to limit the range of the generated notes. Set this to **C4** to **C6** so our melody uses a fairly high register.

For the moment, check '**LEGATO**' - this setting is supposed to make each even play until the next event begins, but as we'll see it doesn't always do this.

Check '**Constrain notes to scale**' and underneath set the scale to match the one we used in the PAD PERFORM part of this tutorial '**C**' '**Natural Minor**'. Now the MPC will only generate notes from the C natural minor scale.

Finally **DENSITY** dictates how close the randomly generated events are to each other, so **50**% is a good starting point. Finally leave **POLYPHONY** to **1** - this ensures that no note overlaps any others, so is essentially 'monophonic' which tends to work well for melodies.

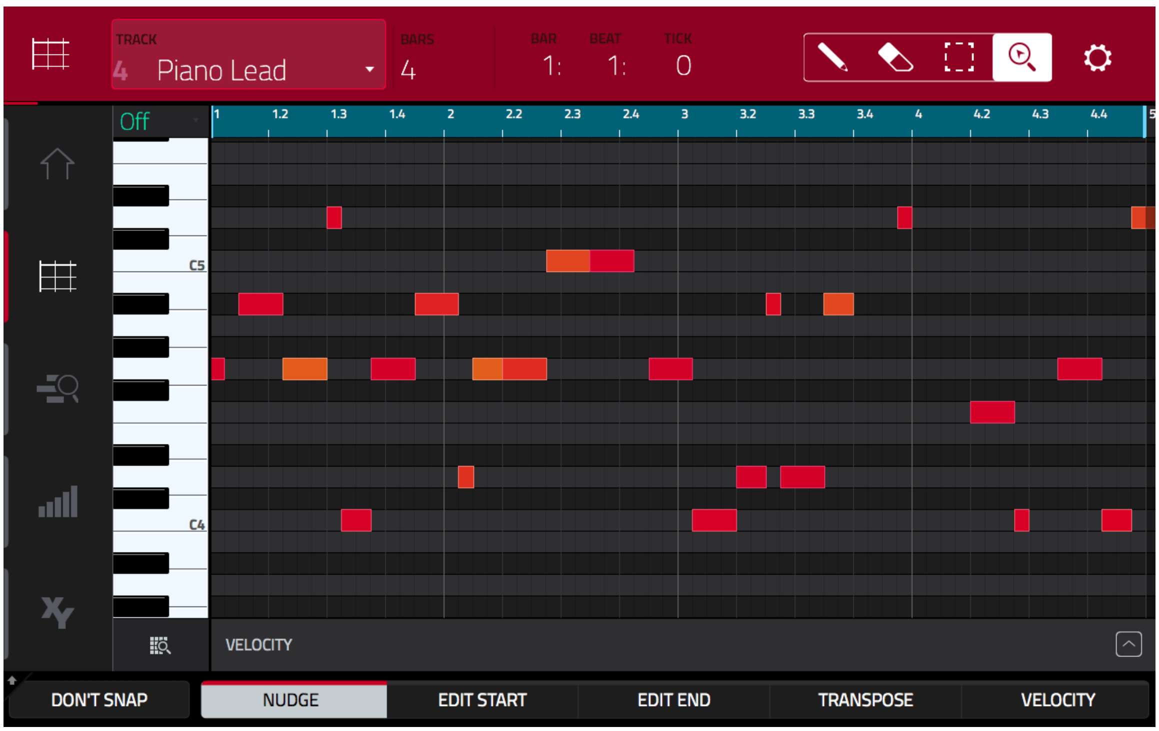
Hit **APPLY** to generate your first batch of notes and hit **PLAY START** to hear them. By hitting APPLY, the 'GENERATE' screen stays active so you can continue tweaking the randomisation algorithm until you find something that works for you.

With your sequence still playing, hit APPLY as many times as you wish until you hear something you like. Due to the random nature of this feature it is unlikely that you'll generate an entire 4 bars of pure excellence; the idea is to use this feature to find a cool snippet within all those random notes.

If you find the notes tend to be a bit sparse, try increasing the **DENSITY** setting, say to **75**%.

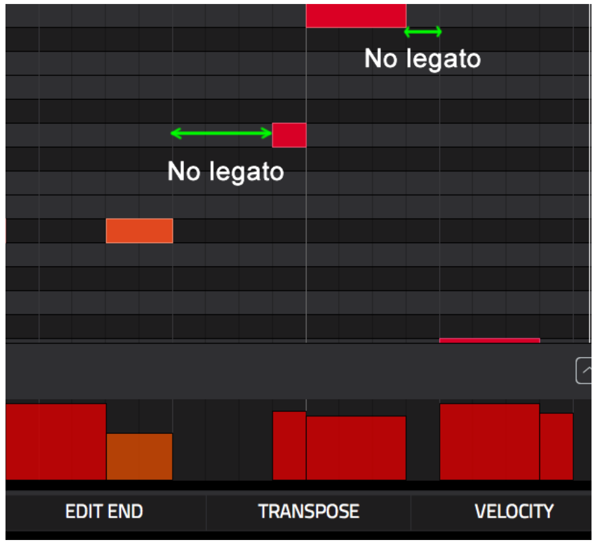
When you do hear something you like, hit **CLOSE**. Whatever you do, don't hit DO IT! Pressing DO IT will generate another set of random notes before closing, wiping over the sequence you wanted to keep.

Go to **sequence 4** (**Random 1-16 Legato**) and select **GRID EDIT** to check out one of my random generations.

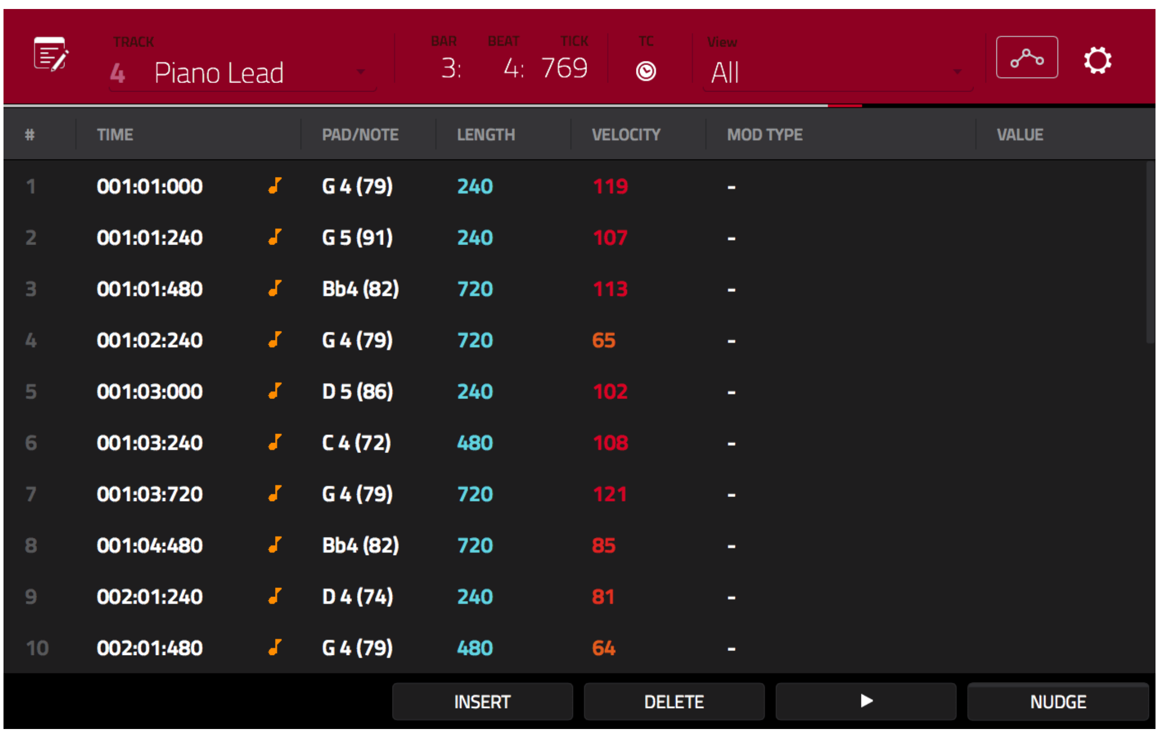


As you can see from all the reds, oranges and yellows, the MPC not only varies the pitch of the events, it also varies the **velocity** for a more natural sounding performance.

The 'legato' setting seems to have worked as each note looks like it plays until the next event, however if you look closely at the grid you can see a couple of notes that don't do this:



Go to **LIST EDIT** and you'll see that each note is placed on a **1/16th** quantise point (even though my T.C. was set to OFF when I generated these notes) and the length (as expected) is a multiple of the NOTE LENGTH quantise 'ticks' - in this case 1/16 which has values of 240, 480, 720 etc.



TRACK 4 Piano Lead — BAR 3: BEAT 4: TICK 769 — TC — View All

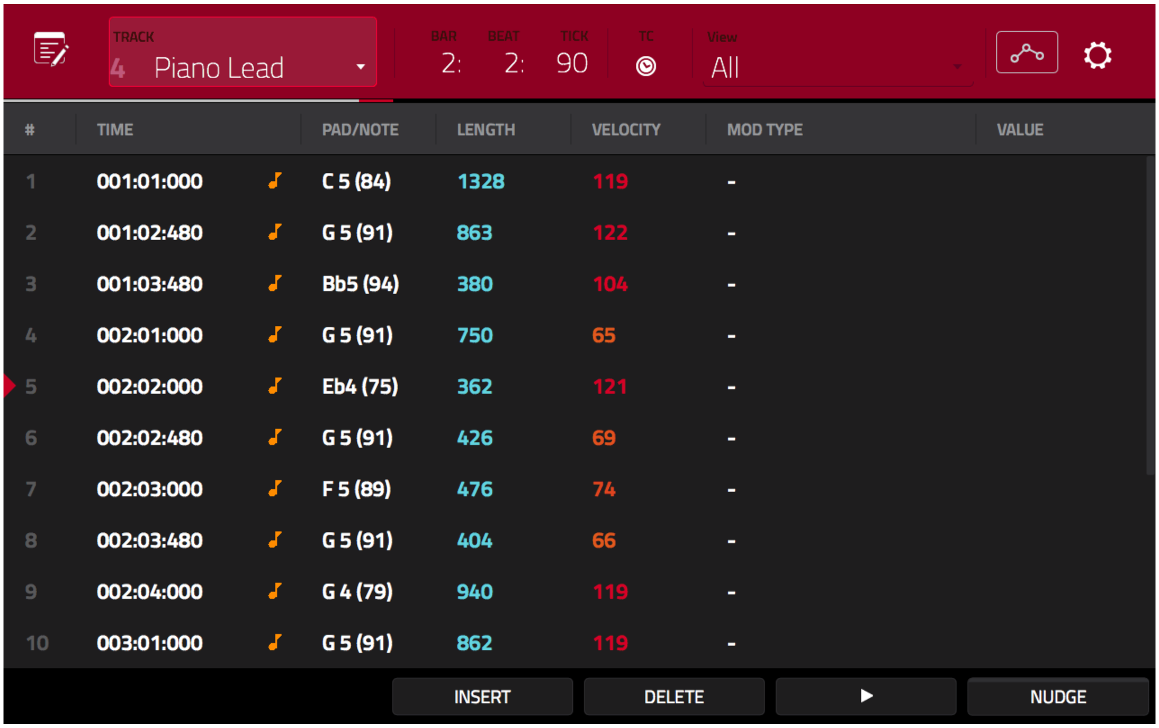
#	TIME	PAD/NOTE	LENGTH	VELOCITY	MOD TYPE	VALUE
1	001:01:000	G 4 (79)	240	119	-	
2	001:01:240	G 5 (91)	240	107	-	
3	001:01:480	Bb4 (82)	720	113	-	
4	001:02:240	G 4 (79)	720	65	-	
5	001:03:000	D 5 (86)	240	102	-	
6	001:03:240	C 4 (72)	480	108	-	
7	001:03:720	G 4 (79)	720	121	-	
8	001:04:480	Bb4 (82)	720	85	-	
9	002:01:240	D 4 (74)	240	81	-	
10	002:01:480	G 4 (79)	480	64	-	

INSERT DELETE ▶ NUDGE

Basically it seems the current T.C. setting is ignored in RANDOM GENERATION mode and the MPC appears to use the **NOTE LENGTH** value as the quantise setting.

Return to **MAIN** and head back to the **GENERATE RANDOM EVENTS** page. This time deselect **LEGATO** and choose a **NOTE LENGTH** of **1/8**. Press **PLAY START** and continue to hit **APPLY** until you hear something you like.

Select **sequence 5** (**Random 1-8 No Legato**) and go to **LIST EDIT**:



This time, with **NOTE LENGTH** set to **1/8** you can see that the events are now only placed at 1/8th quantise points (000, 480, etc), so NOTE LENGTH is definitely being used as the quantise setting. And with **LEGATO** disabled, the 'length' of each event also appears to be random (and not related to the actual NOTE LENGTH setting).

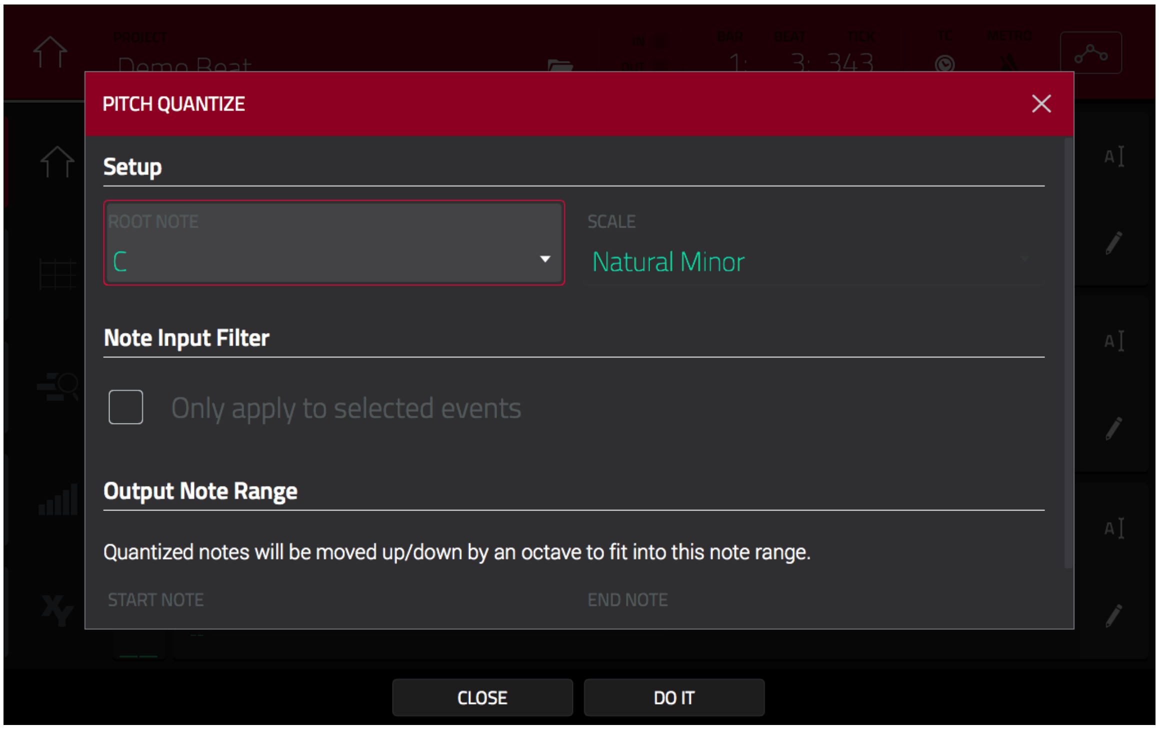
So with your track events randomly generated you can now dive in and start manually editing and tweaking to pull out something usable. For example in sequence 5 I really like the first two bars, bar 3 is okay but I don't like bar 4 at all. Select sequence 6 to hear my edited version.

Using Pitch Quantise

We've already come across 'timing correct' which moves your note events to a specific quantise 'timing' grid, but the MPC can also perform a slightly different type of quantise to musical notes in which the musical pitch of the notes are shifted so they fit into a specific musical scale.

Select **sequence 7** (‘**Bad Performance**’) and hit **PLAY START** to listen to it. Yeah, it’s pretty bad! I recorded this ‘freestyle’ with a T.C. of 1/16, at FULL LEVEL and no ‘pad perform’ in sight. There’s quite a few duff notes in there so let’s automatically fix them.

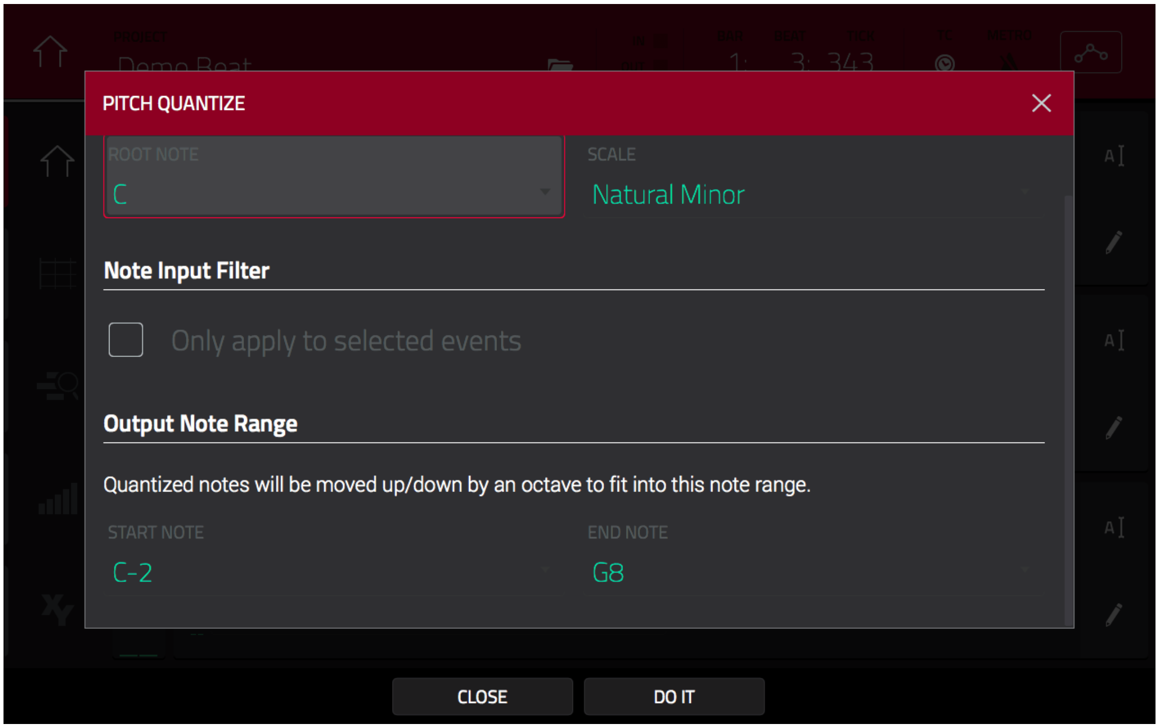
In **MAIN** select the **pencil** icon on the **TRACK** row and choose **PITCH QUANTIZE**.



Set the **ROOT NOTE** and **SCALE** to match the key of our existing sequence, i.e. **C Natural Minor.**

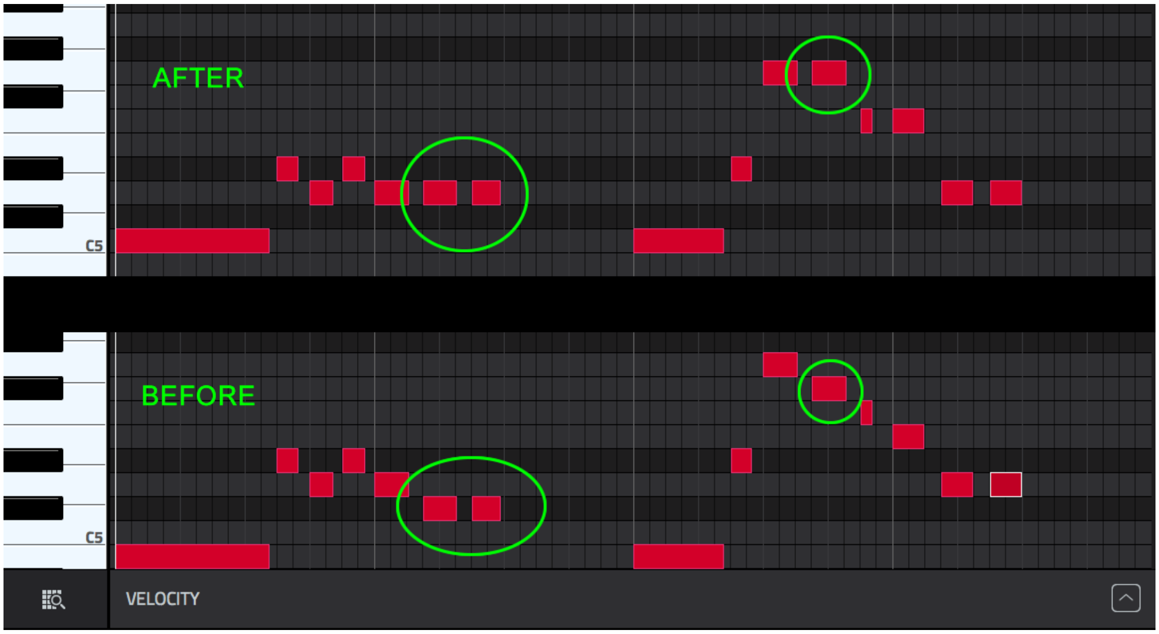
Pitch Quantise can be applied to specific notes by selecting them in the grid first, but in this example we’ll just quantise the entire track so leave ‘**Note Input Filter**’ disabled.

Drag the page up to see the bottom half:



Leave the Output Note Range untouched; we just want Pitch Quantise to move events to their nearest quantise point with no limits on the range that can be used.

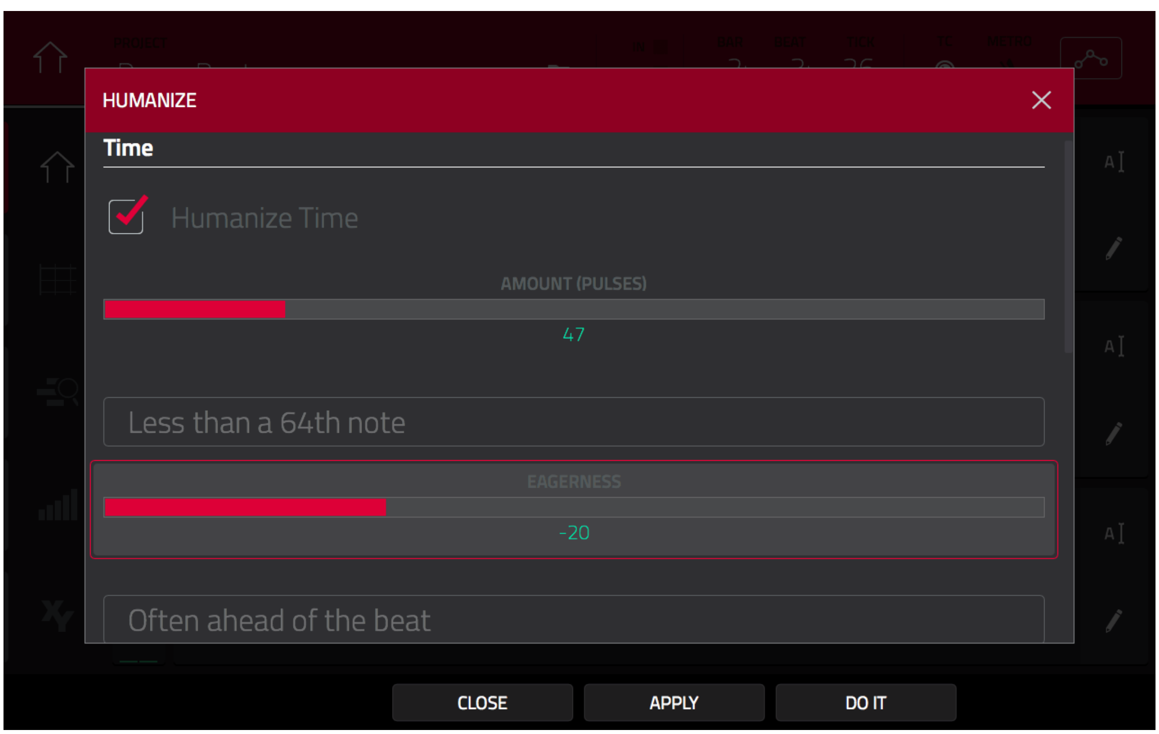
Hit **DO IT** to perform the pitch quantise and hit **PLAY START**; sounds much better! Check out my version on **sequence 8** (**Pitch Quantize**). If you compare the sequences via GRID view you can see how the MPC has shifted some of the notes to the nearest note in the C Natural Minor scale (I've just highlighted a couple of the example below):



Using Humanize For a More Natural Feel

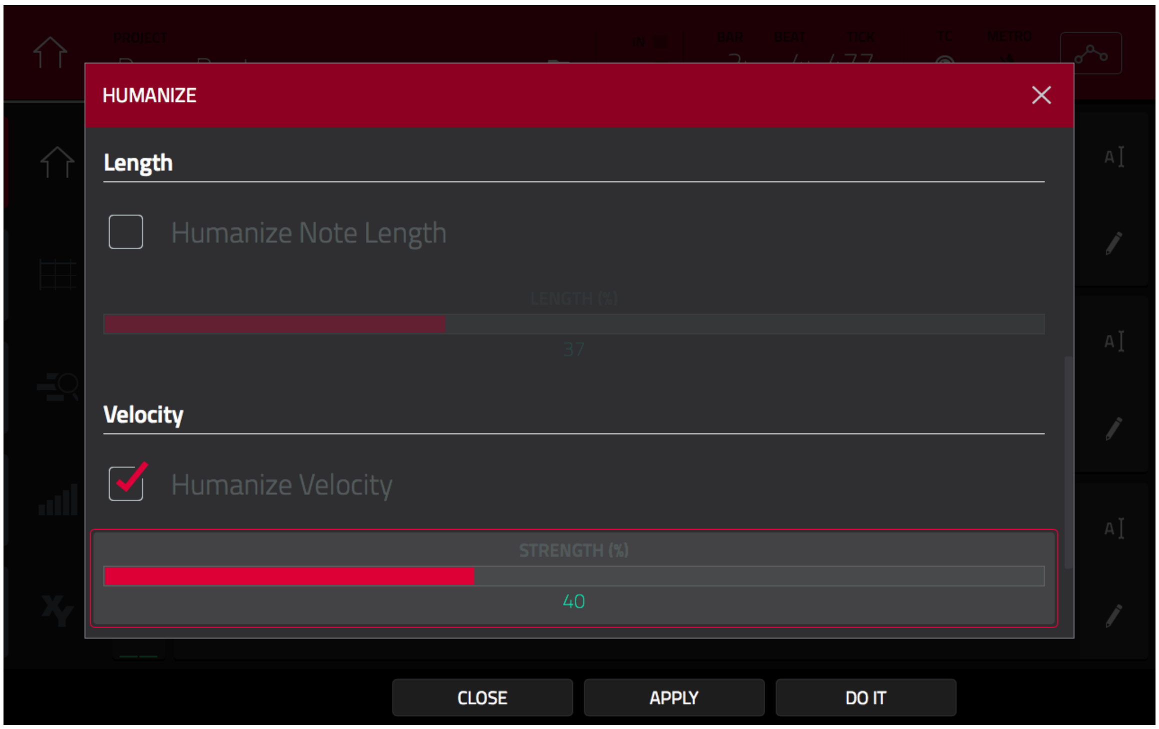
A natural piano performance will normally feature a variance in velocity and a less than rigid timing, so to 'loosen up' our sequence we can use the HUMANIZE feature.

From **MAIN** select the pencil on the **TRACK** row and select **HUMANIZE**:



Make sure '**Humanize Time**' is selected, this is going to shift our events so they lose that rigid feel. There are two parameters that control this time shift; AMOUNT and EAGERNESS. In the above example I'm using quite subtle time settings, with the EAGERNESS set to produce events that will put the performance slightly 'ahead' of the beat (as indicated in the help tip on screen).

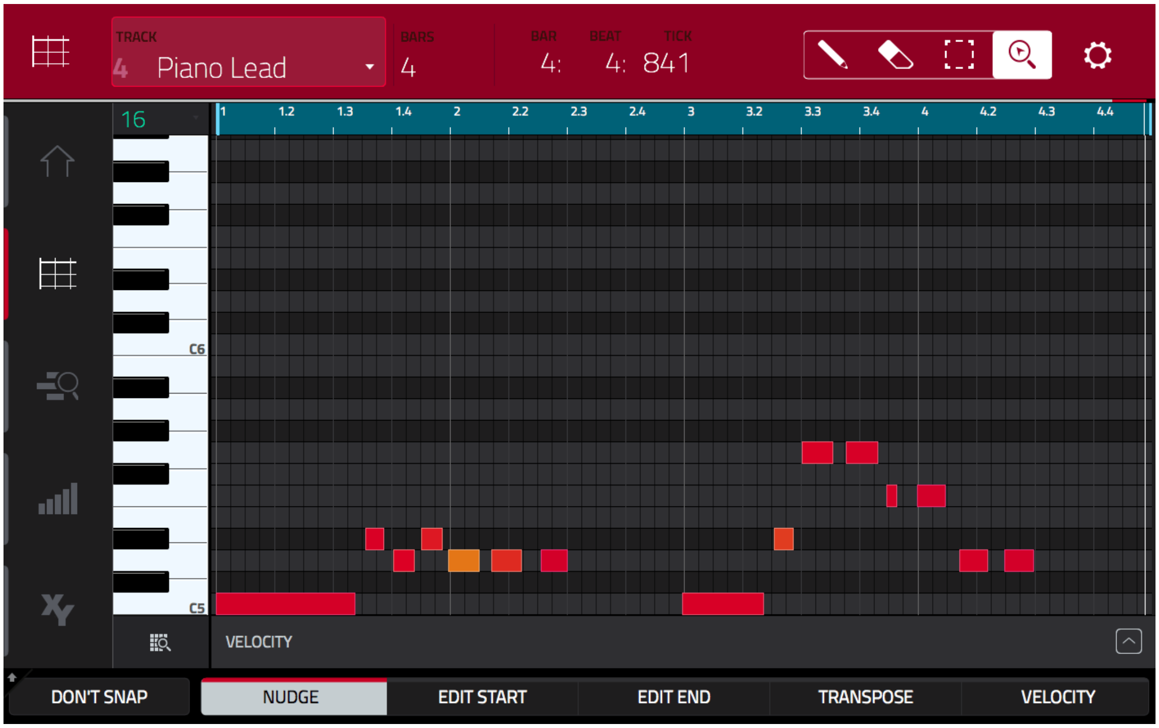
Scroll up to see the next settings.



Leave '**Humanize Note Length**' unchecked, we'll look at his option in a bit.

Finally check '**Humanize Velocity**'; I've used a moderate amount here (40), this will bring some decent dynamics to the performance.

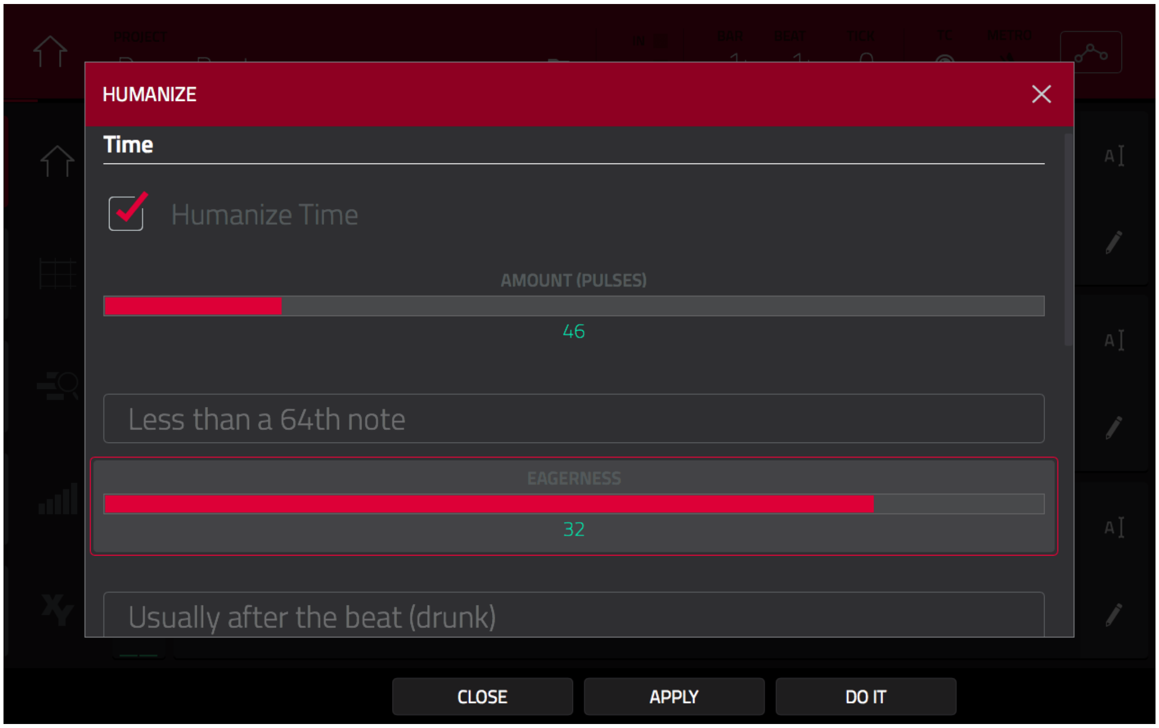
Hit **DO IT** and go to **GRID**:



It's immediately clear that we now how some good variations in dynamics, wth randomisation that you wouldn't achieve when manually drawing in a new velocity curve within the velocity lane (see chapter 4).

Press **PLAY START** and you should hear that the performance is slightly ahead of the beat, and you can see this in the GRID as many events now occur a few ticks shy of the original 1/16 quantise points.

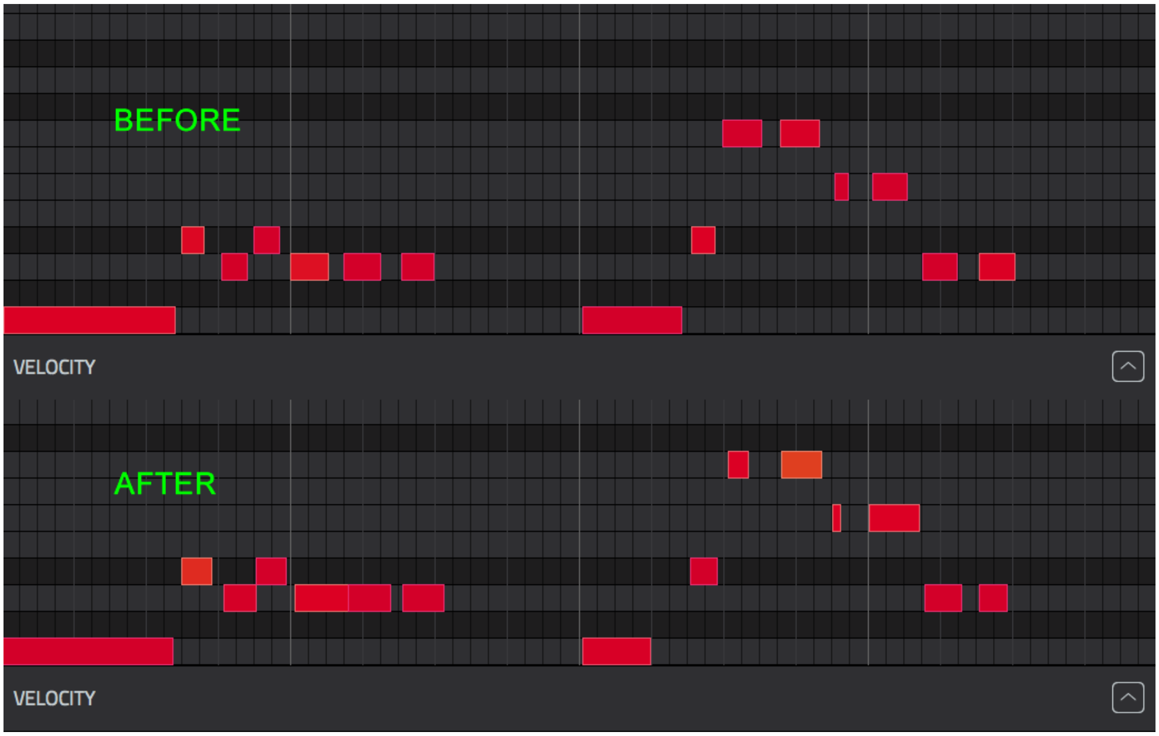
Hit **UNDO** to revert the sequence back to it's 'rigid' original and select **HUMANIZE** once more.



This time let's just change the **EAGERNESS** setting to a positive value, which as the help tips states, is going to set the performance to be slightly after (behind) the beat. Hit **DO IT** and **PLAY START**. This time the feel is more lazy and relaxed, with some events now occurring slightly after the original 1/16 quantise points.

Check out **sequences 9** and **10** for my versions - switch between the two to compare the subtle difference in feel.

Select **sequence 11** (**Humanize Length**) - in this sequence I applied the same settings as sequence 10, but I also included **Humanize Note Length** with a setting of **50%**. Compare the note lengths in GRID:

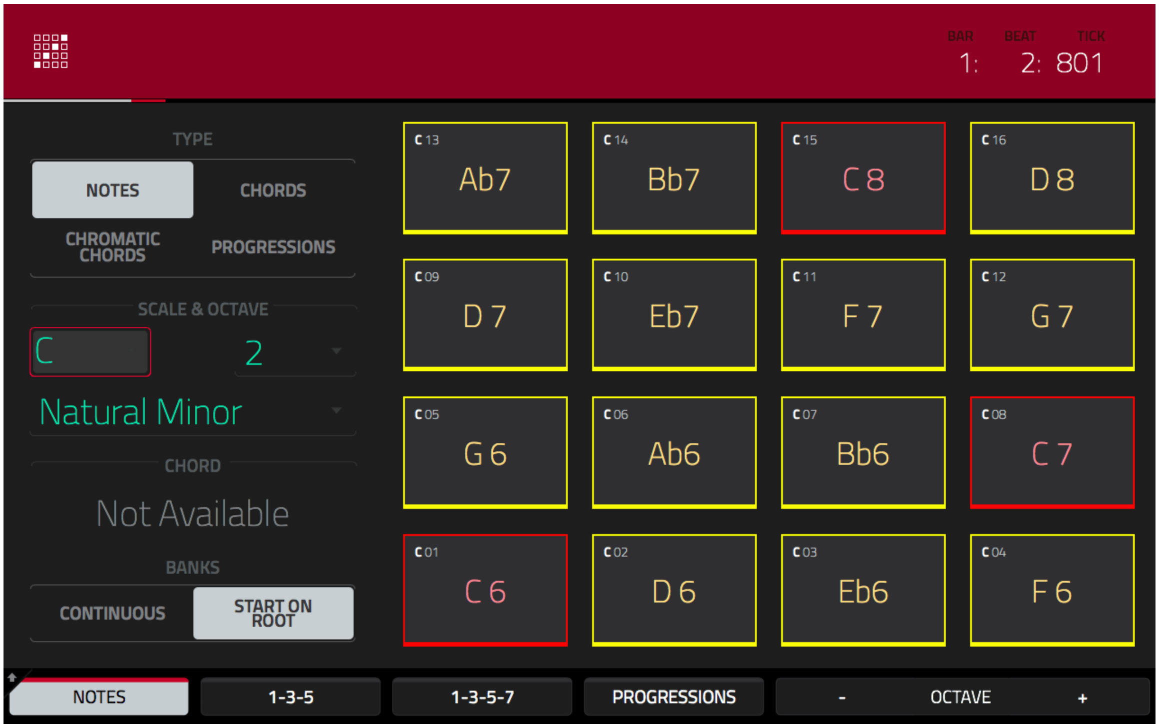


You can see that some notes are longer, others are slightly shorter, giving a nice, naturally sounding variation to the events. Humanize is a great feature, use it wisely though, I'd recommend sticking to subtle changes where possible, but remember you can always use UNDO to revert back to the original sequence, or make a copy of the sequence first (or copy the individual track) and experiment with the copy.

Pad Perform 'NOTES'

We used Pad Perform earlier to create our piano rhythm, but Pad Perform is not just limited to chord generation, it can also be used to play melody lines.

Select **sequence 12** which contains a blank track 4, ready for recording. Head back to **PAD PERFORM** and set **TYPE** to '**NOTES**', keeping **SCALE & OCTAVE** at **C Minor**. Finally set the '**BANKS**' parameter to '**START ON ROOT**'. Select **BANK C**.



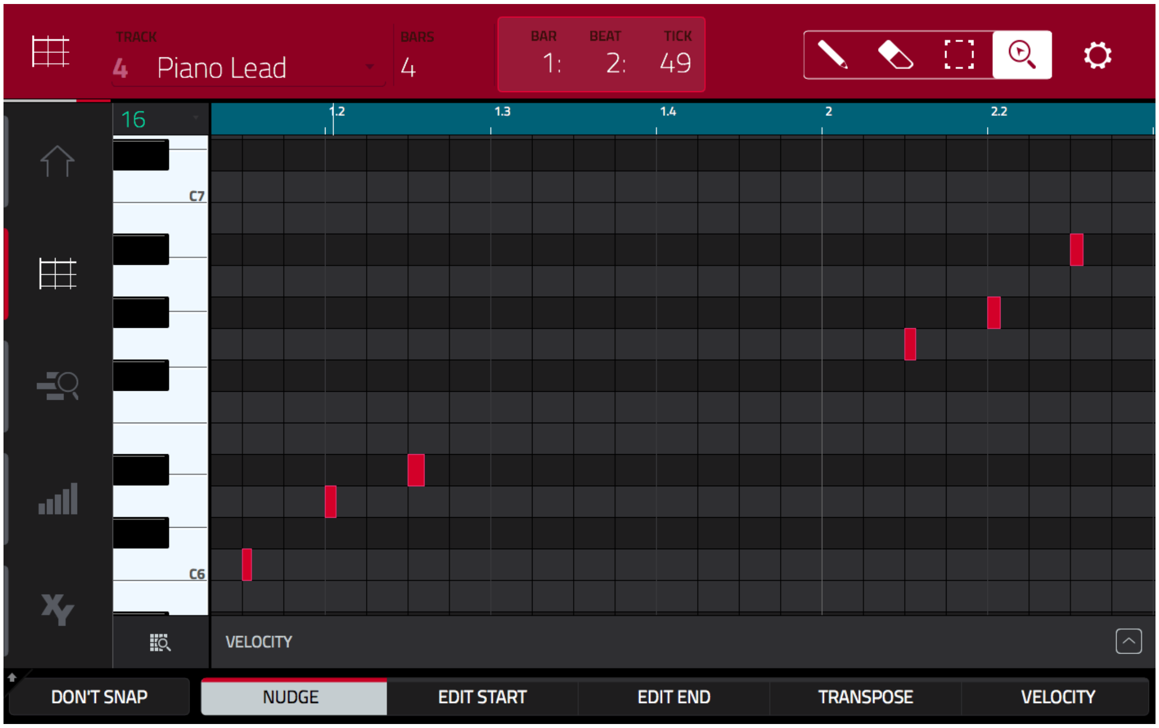
When you have set pad perform to '**notes**' mode, the MPC sets your pads to only play notes from the specific scale you set under '**Scale**'. So in this example, we set a scale of '**C Natural Minor**' (to match the piano chords and bass line), which consists only of the notes **C, D, Eb, F, G, A, Bb, C**. It is not possible to play any other note outside of this C minor scale, so there's no chance of playing a 'duff' note.

Notice how most of your pads are illuminated in yellow expect for three (pad 1, pad 8 and pad 15), which are red. These are called the '**root notes**', which in the case of **C** minor, are **C** notes, each one an octave higher than the previous one.

Play your pads from C1 to C16. To move your notes up an octave, you can press **OCTAVE+** (at the bottom right of the screen). The currently available notes are shown on the on-screen grid; and as 'Start on Root' was selected, the first pad of any bank is always a root note (in the current scale, C).

If you prefer the notes to continue in progressive order across banks, just choose '**Banks: CONTINUOUS'.**

Use Pad Perform 'notes' to record a melody to your beats – keep it simple, just add a nice little repetitive hook. Select **sequence 13** (**Piano Melody**) to hear my version. In this I have kept the melody very simple using 'arpeggios' which are effectively 'broken chords' that follow the underlying chords of the sequence (C minor > G minor). This was easily achieved in PAD PERFORM 'NOTES' by hitting pads C01, C02, C03 followed by C05, C06, C07.



Choosing Other Instruments

So far we've used a KEYGROUP sample program as our MIDI sound source, but as I mentioned earlier in this chapter, we can use the recorded MIDI data on tracks 3 and 4 to trigger sounds from *any* MIDI instrument.

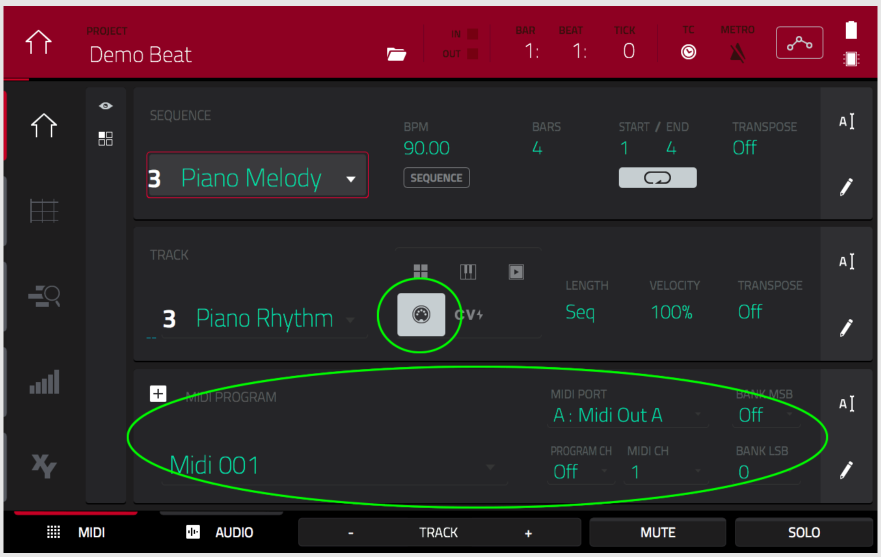
For example, the piano program we've been using is ultimately quite basic, but we can replace it with a more complex keygroup program at any point. For example, I created the **Concert Grand** ([http://www.mpc-samples.com/product.php/257/concert-grand-mpc-expansion-pack/)](http://www.mpc-samples.com/product.php/257/concert-grand-mpc-expansion-pack/) which features a keygroup program that contains 4 layers of chromatic keygroups for a very realistic piano recreation. You could load this into memory and assign it to the two piano tracks and the MPC would now lay the exact same sequence performance but this time triggering the more realistic sounds from the Concert Grand. You could of course even use completely different instrument programs such as a multisampled Rhodes or B3 organ for example.

But we're not just limited to keygroup programs. You could change the track TYPE to **MIDI** and trigger the sounds from an attached MIDI sound module or synthesiser. And if you are in 'controller' mode you have another option – you can assign a virtual 'software instrument' (PLUGIN) to the track and trigger any patches or instruments available to that plugin.

Triggering a MIDI Instrument

To trigger external MIDI sound modules you'll need to use a track '**TYPE**' of **MIDI**. You can even change the track type of existing tracks, so for example instead of triggering the Piano Program on tracks 3 and 4 of our current sequence we can change the track type to MIDI and send the existing MIDI events we recorded over to a connected MIDI sound module.

To choose a MIDI track, hit the **MIDI** icon on the **TRACK** row in **MAIN**:



Here you can see on the **PROGRAM** row that a MIDI track has different configuration options to a KEYGROUP program. The two key parameters are **MIDI PORT** and **MIDI CH**.

MIDI PORT tells the MPC which physical MIDI output port to send out the MIDI information to. The MPC Live has two MIDI OUT ports (A and B), while the MPC X has four MIDI OUT ports (A-D). Simply set MIDI PORT to correspond to the MIDI OUT you have connected your MIDI instrument to – if your MIDI instrument is connected to MIDI OUT A, set **MIDI PORT** to **A**.

MIDI CH tells the MPC which **MIDI *channel*** to send the MIDI information out on. There are 16 channels of data available to each MIDI cable and your receiving MIDI instrument can be configured to route each incoming MIDI channel to a different internal sound patch. If the piano patch on your sound module is set to receive information on MIDI channel 3, then set **MIDI CH** to '**3**'.

> Once configured correctly, your MPC will now send the MIDI events recorded on these tracks to your attached MIDI sound module. Please refer to **Appendix A** for further information on attaching MIDI instruments to your MPC.

The key point is that there are fundamentally no differences between the way the MPC ultimately records the data to these different types of 'instrument' tracks – whichever type of instrument source you choose the end result will be a track of standard MIDI events that will trigger the sounds from whatever MIDI instrument is 'attached' to that track, be it a MIDI sound module, a keygroup program or a software plugin.

The advantage of keygroups is that they are always available to us whether we are working with a computer or using the MPC in 'standalone' mode and they are completely 'in the box'. However keygroups tend to use up a lot of sample memory and CPU resources.

Using Software Instruments In 'Controller Mode'

If you do work with your computer, software instruments definitely provide you with a vast range of sounds. You can install most VST and AU plugins inside the computer version of the MPC Software, including all leading synths, sound modules and even Kontakt which in turn can be used to play back incredibly complex multisampled instrument libraries.

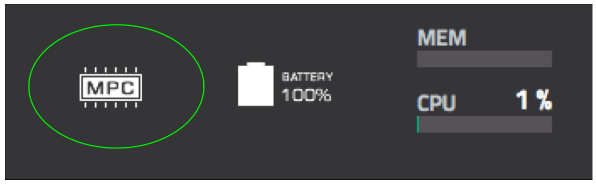
But even if you don't have any third party plugins installed, you'll definitely have one software plugin available inside the core of the MPC Software; **Hybrid 3**. This is basically a self contained software-based sound module installed in your computer which contains many different instrument sounds (patches) that can be loaded into an MPC Project

It's important to understand that plugins are NOT available while you are in standalone mode – these are a '**controller mode'** only option. Even the integrated Hybrid 3 instrument is unavailable when in standalone mode.

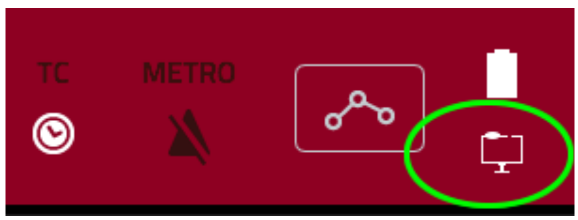
So while this book is focussed on the more 'standalone ' aspects of the new MPCs, I wanted to briefly show you how to record a plugin based track in your computer – don't worry, once recorded we're going to convert this plugin track into an audio track that we can use in standalone mode!

Configuring a Plugin Track

Firstly, connect your MPC to your computer using a USB cable and fire up the MPC Software application. In your MPC enter **controller mode**. To do this go to **MENU** and tap on the **MPC chip icon**:



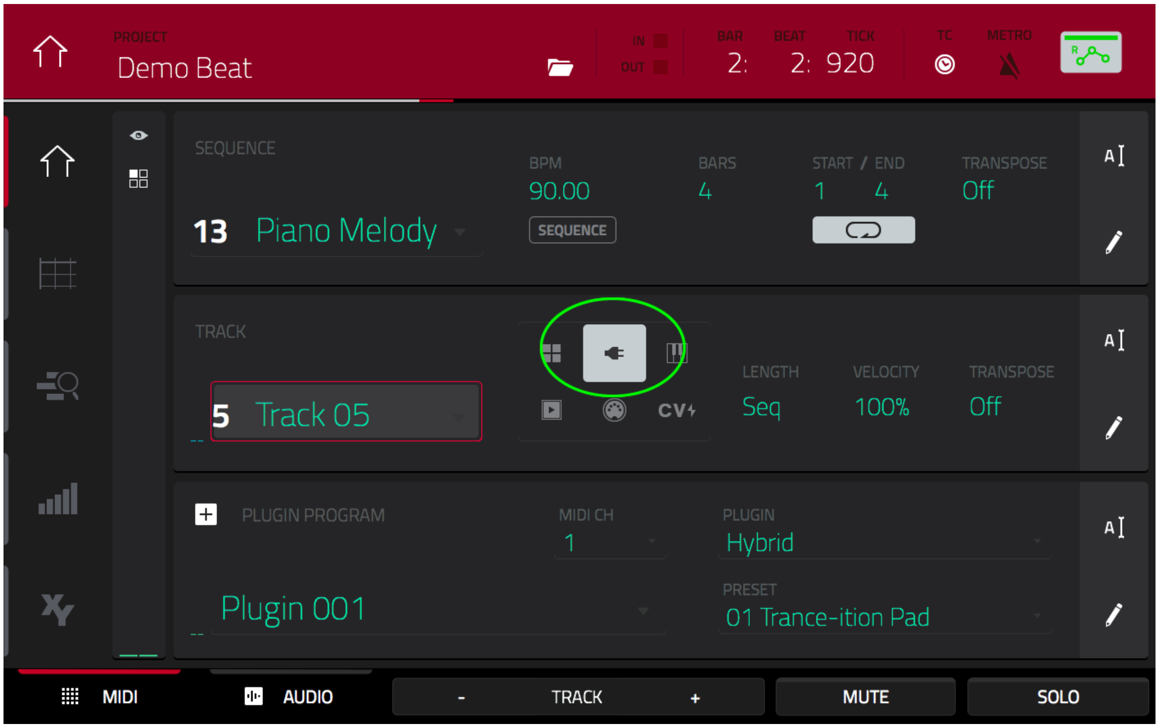
Now tap on the **CONTROLLER MODE** button to enter controller mode. At this point you should be able to continue using your touchscreen to control the MPC, but you are now controlling the MPC Software application on your computer. You know you are in 'controller mode' by the computer monitor icon in the top right of your screen:



Problems Entering Controller Mode?

Sometimes if the MPC Software was already opened in my computer, my MPC refuses to connect and instead remains on the 'Looking for Computer' screen. The solution is to close the MPC Software and re-launch it. Alternatively, do not launch the MPC Software at all until you have pressed the CONTROLLER MODE button in your MPC.

In your MPC hardware, with sequence **13** (**Piano Melody**) selected, go to **MAIN** and select the unused **track 5** and tap on the **PLUGIN** icon.



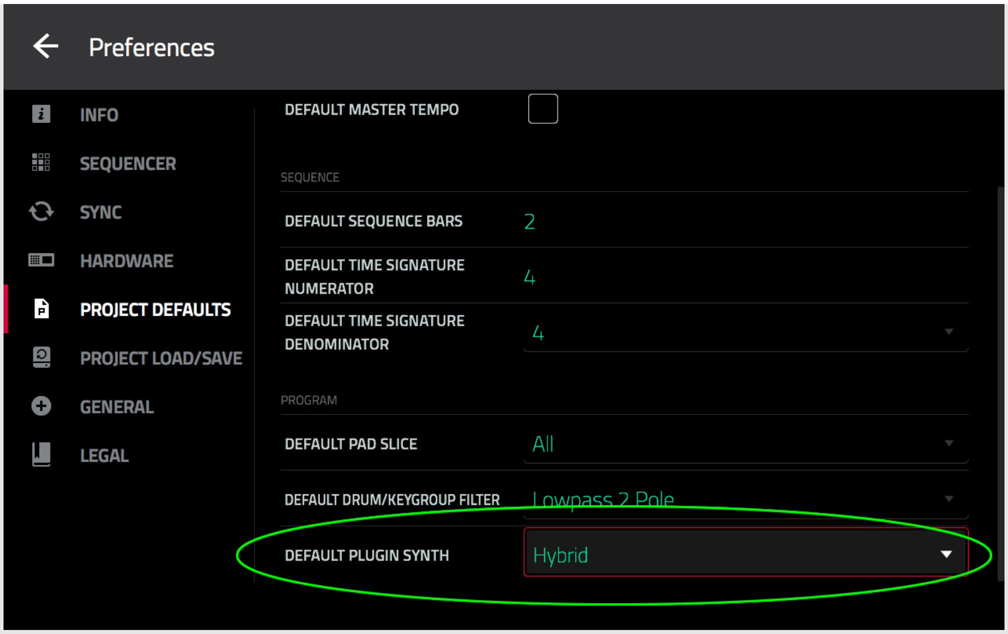
Track 5 is now assigned as a **PLUGIN** type track and is ready to accept any compatible *instrument plugin* as its sound source. The MPC has even created a plugin program (**Plugin 001**) for it which you can rename if you wish, just like any other program type.

A PLUGIN type track is very similar to a KEYGROUP track. However instead of sending MIDI events to a keygroup sample instrument, a plugin track sends MIDI events to a *software instrument* installed on your computer.

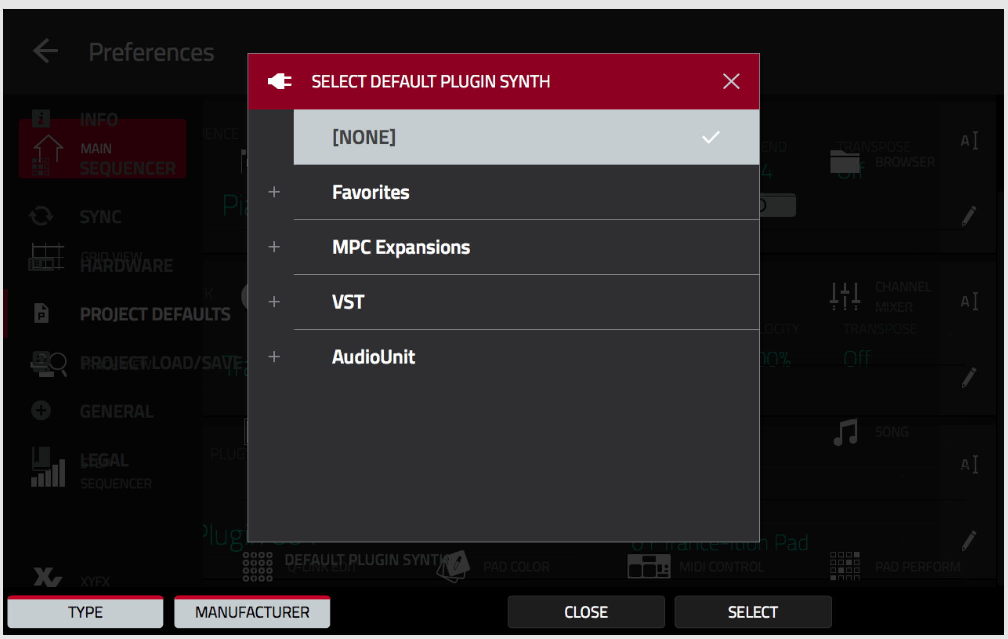
In a default installation of the MPC Software, the MPC will automatically assign the built in **Hybrid** synth plugin (as indicated under the **PLUGIN** parameter) which is perfect for this tutorial as it's a plugin available to all MPC Software users.

Changing The Default Plugin

Go to **Preferences > PROJECT DEFAULTS > DEFAULT PLUGIN SYNTH**:



Here you can set a different synth to automatically load on a new plugin track. To change the default 'Hybrid' synth, double tap '**Hybrid**' to bring up the **Plugin Manager**:

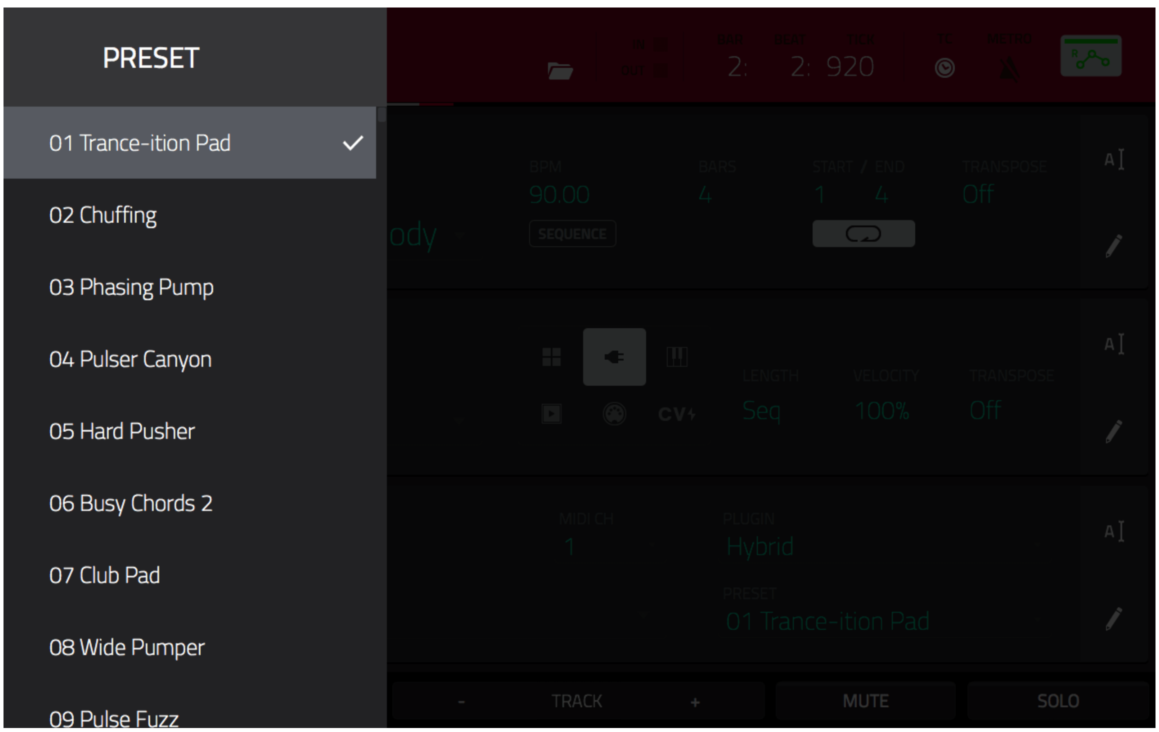


Here you will see a list of available third party plugins or MPC expansions that can be assigned to our plugin track. As you can see, this is very similar to the Effects Manager we discovered previously in the book, and it will use the same 'Sort By' options that you chose when dealing with effects (TYPE and MANUFACTURER).

However this time this dialog only shows *instrument* plugins, nothing else. To select a default plugin, simply locate it from the list of available plugins and double click it. If you'd prefer the MPC to not assign any plugin synth automatically, just select **[NONE]**.

Referring back to your track information panel, you should see that the initial Hybrid patch loaded is '**Trance-ition Pad**' – hit some pads in various pad banks to hear how it sounds.

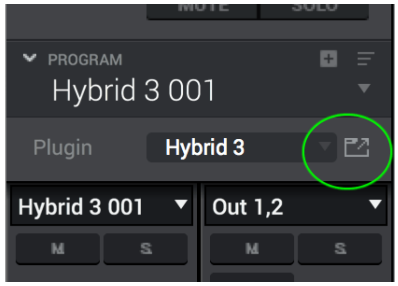
Now we actually need a bass sound, so let's find a nice bass preset patch within Hybrid 3. One way to select a preset is to double tap the '**PRESET**' dropdown and select one from there:



As you scroll down you'll soon realise that there are hundreds of patches and none are categorised, so it's going to be a bit of a slog finding a particular type of patch sound.

The Hybrid 3 Plugin Interface

For a more intuitive method of patch selection, let's open up the plugin itself, so this is one of those rare occasions in this book where we head over to the computer GUI. To open the plugin interface, make sure the ‘Inspector’ is enabled (press ‘I’ if it isn’t) and click on the box/arrow icon next to the plugin name:



You’ll now see the **Hybrid 3** plugin interface:

Here you can choose any patch and then optionally tweak the various parameters to create a unique sound to your own liking. One benefit of working in the plugin interface is that the patches are nicely organised into sound type. In the top right of the plugin, click on the **’01 Trance-ition Pad**’ patch name:

Here you can see all the different patch categories, with an '**X**' indicating the currently selected category. Click on **'08 Arpeggios'** and select **'09 Thin To Thick Modwheel'**. Hold down a pad and you'll hear a nice arpeggiated chord play, which will be in perfect sync with your sequence tempo.

Recording a Plugin Instrument

Enter **PAD PERFORM** mode and as we did with the piano lead, select **NOTES**, with **C NATURAL MINOR**, **OCTAVE 2**. Choose pad **BANK B**:

Hit **REC** and **PLAY START** and record some arpeggiation. I kept this simple but effective – press and hold pad **B01** (a C note) for the entire **first bar** (and **third bar**), then press and hold **B05** (a G note) for the entire **second bar** (and **fourth bar**).

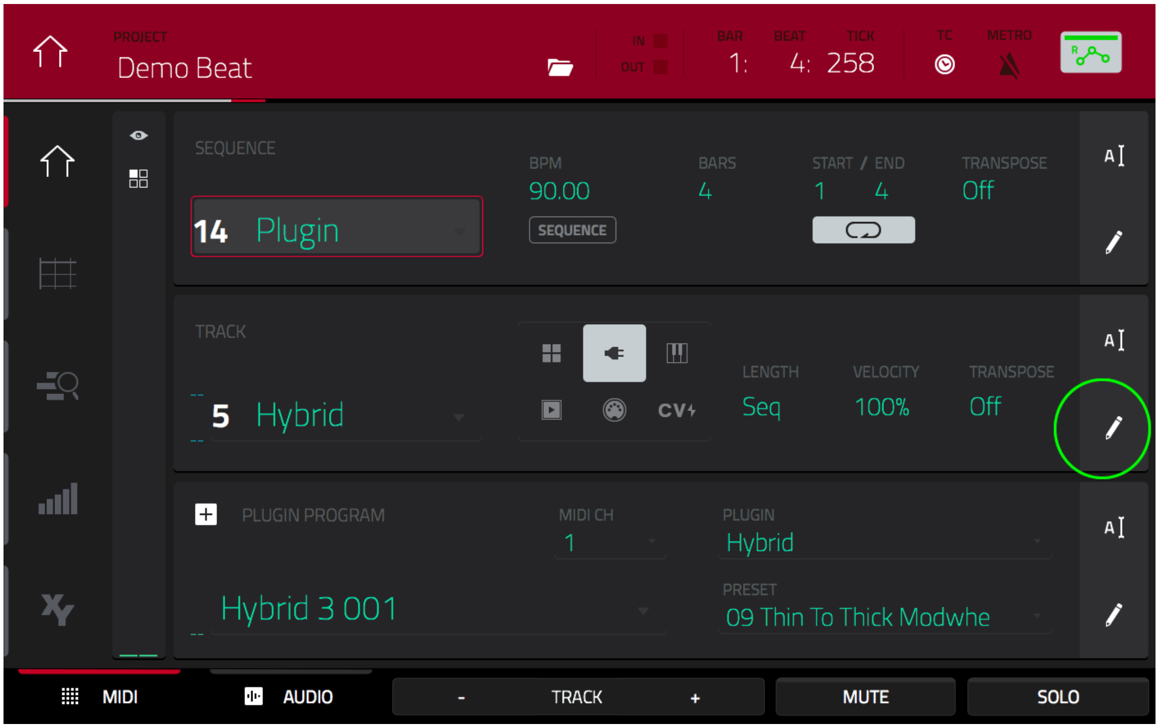
Listen to my version by loading the project **Demo Beat Plugin.xpj** from the chapter 11 folder – if you want to just hear the plugin track, select **track 5** (**Hybrid**) and hit the **SOLO** button; this will temporarily mute all other tracks. This track is really starting to come together with lots of nice layers and harmonies.

Bouncing Plugin Tracks To Audio

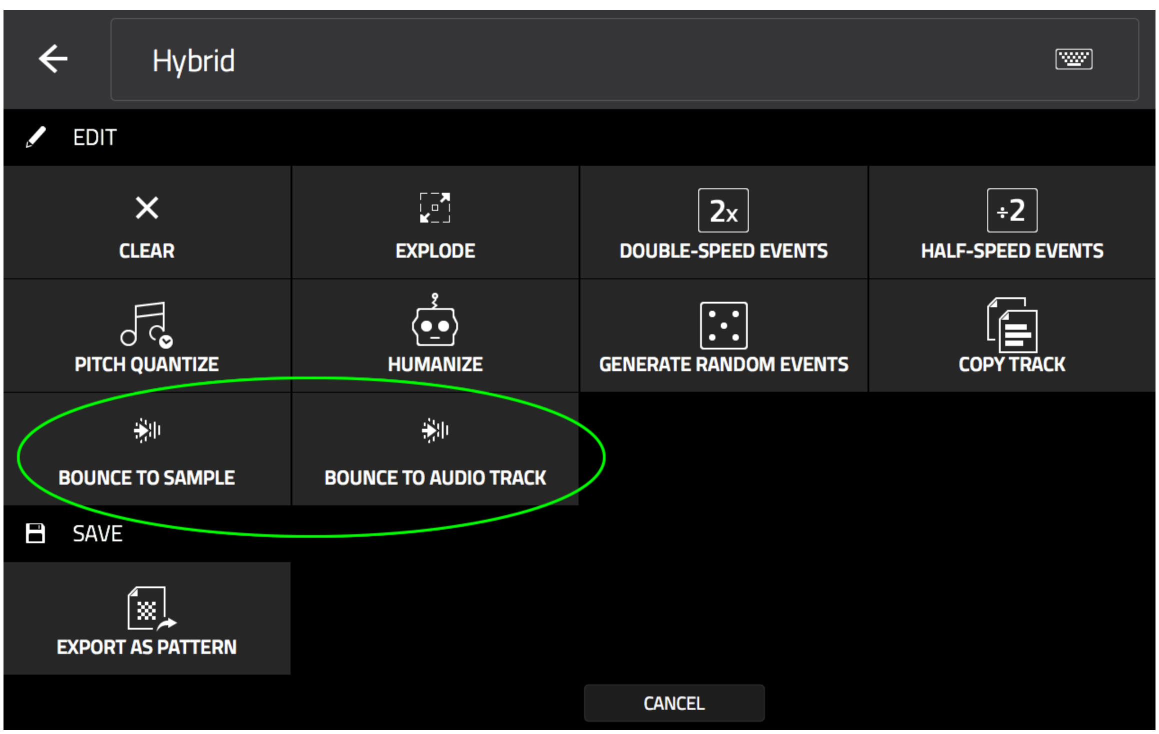
Plugin tracks are great but do require you are ‘tethered’ to your computer so present a problem for MPC Live and MPC X owners who want to continue working on a project in standalone mode.

Additionally, many plugins are resource hogs, eating up computer CPU, which can become a very real issue if you have a project that utilises many different plugin tracks – this is true for all MPC controllers.

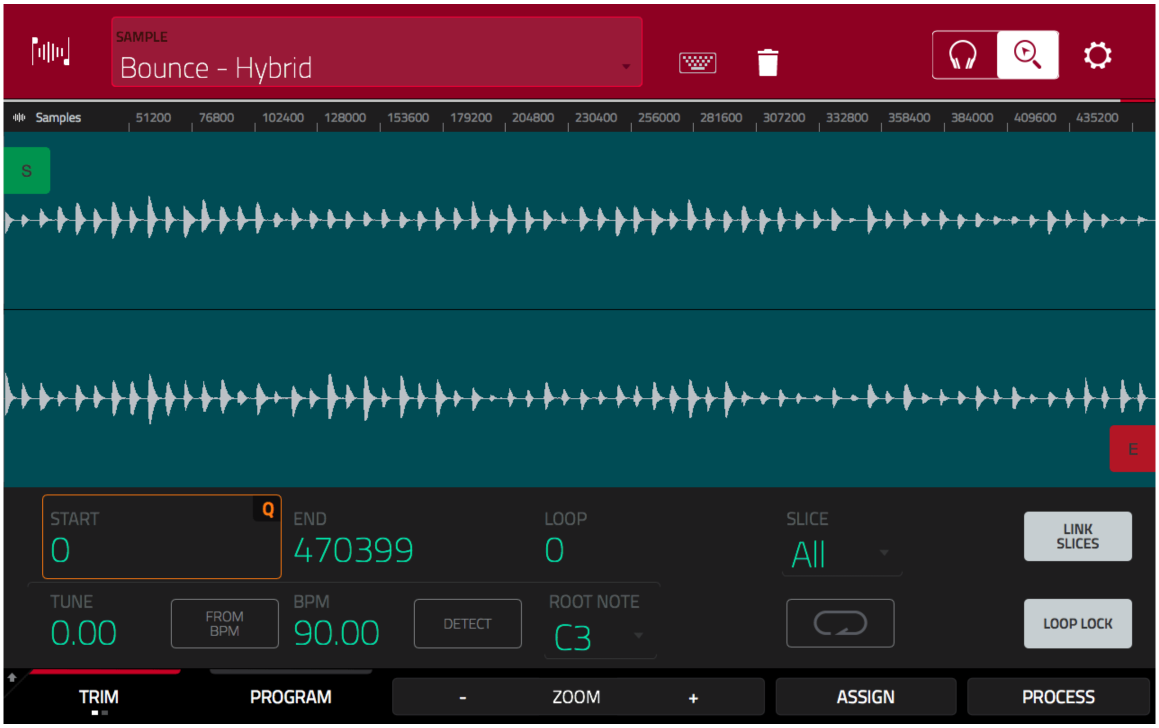
There is however a solution to these problems. In the current ‘Plugin’ sequence, select **track 5 Hybrid** and on the **TRACK** row, click on the **pencil icon**:



At the end of the **EDIT** row there are two options; **BOUNCE TO SAMPLE** and **BOUNCE TO AUDIO TRACK**.



Both of these options will create a ‘bounce’ of our plugin track. This bounce is basically an audio recording of the plugin track. Click on **BOUNCE TO SAMPLE** to create a bounce of our Hybrid track. Now go to **SAMPLE EDIT** and locate our bounced track – **Bounce – Hybrid**.



Hit **pad 1** to preview our bounce. As you can see, the MPC has just recorded the outputted audio from the plugin track and created a sample (named after the original track name). This is a standard stereo sample which can be edited and chopped like any other sample and assigned to a DRUM program (if you wish).

However the second option we originally had, '**BOUNCE TO AUDIO TRACK**', provides an even better solution as it instantly assigns the bounced sample to an MPC audio track. Hang on, what's an audio track…?

012 First Look: Audio Tracks

So far we've only been dealing with MIDI tracks and MIDI event data, but the MPC Software also has another core track 'type'; the **audio track**. Unlike a MIDI track that only consists of a series of MIDI events used to trigger sound sources, an audio track contains nothing but actual audio.

If you've used a digital audio workstation (DAW) like Logic, FL Studio, Pro Tools or Ableton you'll no doubt already be very familiar with the concept of audio tracks. Audio tracks can be used more like a traditional multitrack recorder or tape recorder, typically used for recording 'non-MIDI' performances such as vocals, acoustic and electric guitars, string instruments like violins, brass instruments, turntable scratching, live drum kits and so on.

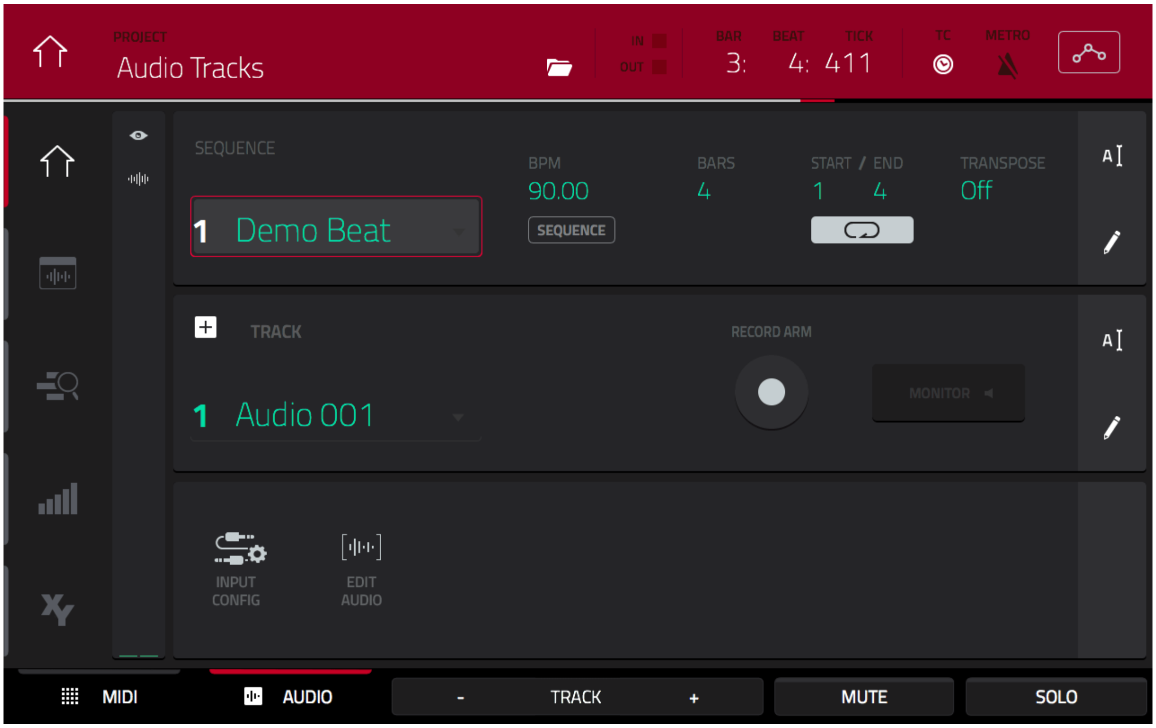
Later in the book we'll look at recording a vocal performance directly to an audio track, but at this point let's just import an existing recording into an MPC audio track so we can become more familiar with the workings of audio tracks within the MPC sequencer.

Setting Up an Audio Track

At the end of the last chapter I showed you how to create a melody line with the Hybrid plugin in 'controller mode' – we then 'bounced' this plugin track and created a WAV version of it; **Bounce – Hybrid**. By bouncing down the plugin track we are able to bring the audio generated by a 'controller mode' plugin back into the 'standalone' environment.

Let's start afresh by loading the project **Audio Tracks.xpj** from the **chapter 012** folder. This will load up our **'Demo Beat'** into **sequence 1** and will also load the **Bounce – Hybrid** sample into project memory. I've left the Hybrid 3 plugin track (5) there, but it is muted (and regardless it will not work in standalone mode either way); in fact I have converted it to a MIDI track otherwise you'd be getting on-screen warnings regarding how you cannot use plugins in standalone mode.

In **MAIN**, press the **AUDIO** button at the bottom of the screen:

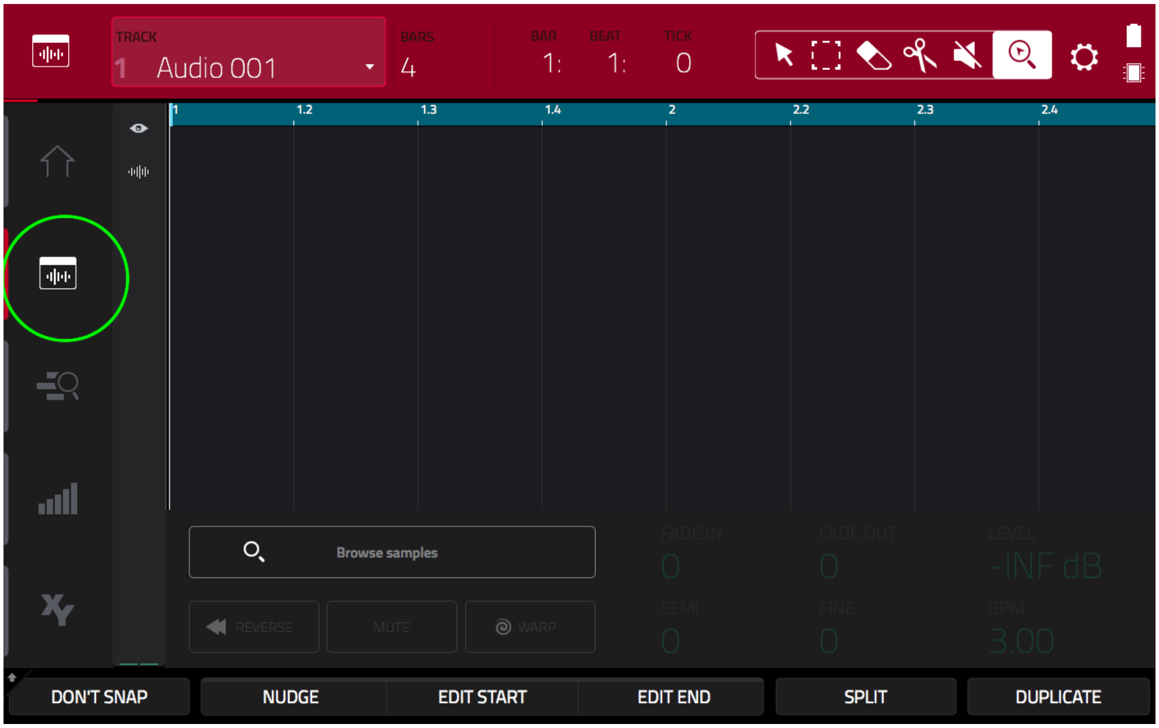


This has changed the currently active core track type in the MPC sequencer from MIDI to **AUDIO**. In each sequence, audio tracks exist *separately* to MIDI tracks, so in terms of audio tracks, we are currently only on track 1 (while we have currently used 5 different MIDI tracks).

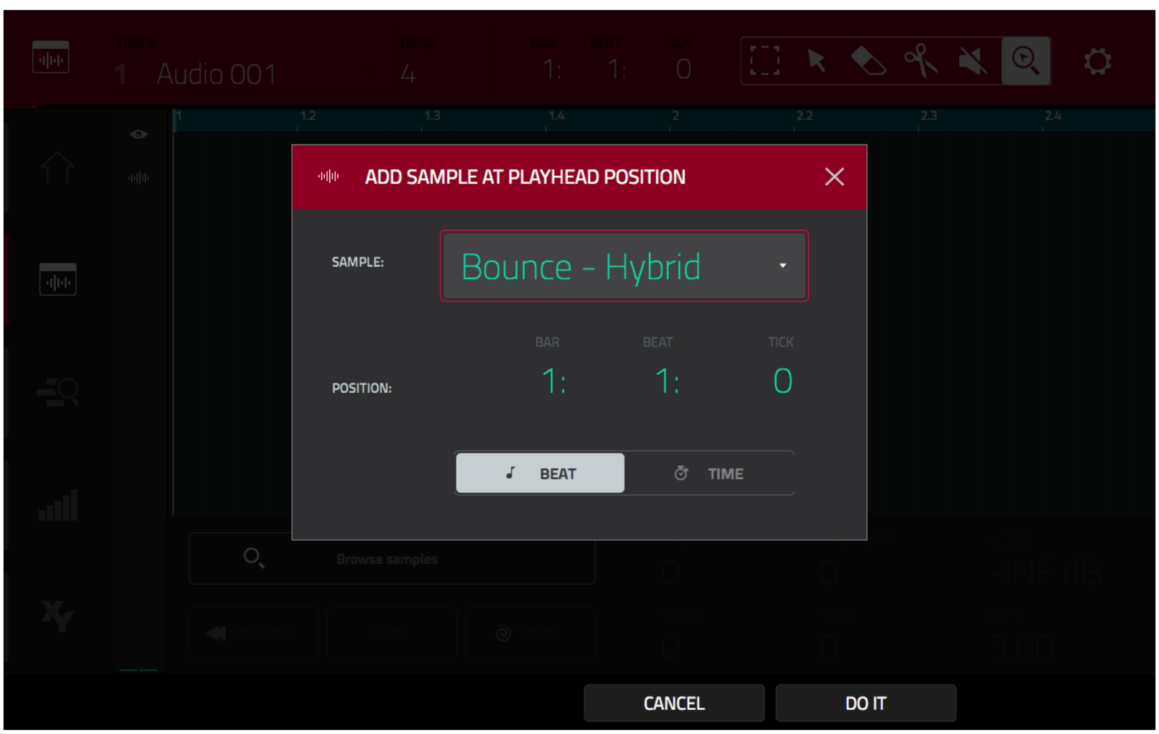
The top row of MAIN is unchanged and continues to display the core sequence configuration (tempo, length etc). The middle row shows our track name (Audio 001), a number of editing options (via the pencil icon) and a '**RECORD ARM**' option.

At the moment our Audio 001 track is empty – later in the book we'll look at recording audio directly to an audio track, but at this point let's just import an existing audio recording to this track.

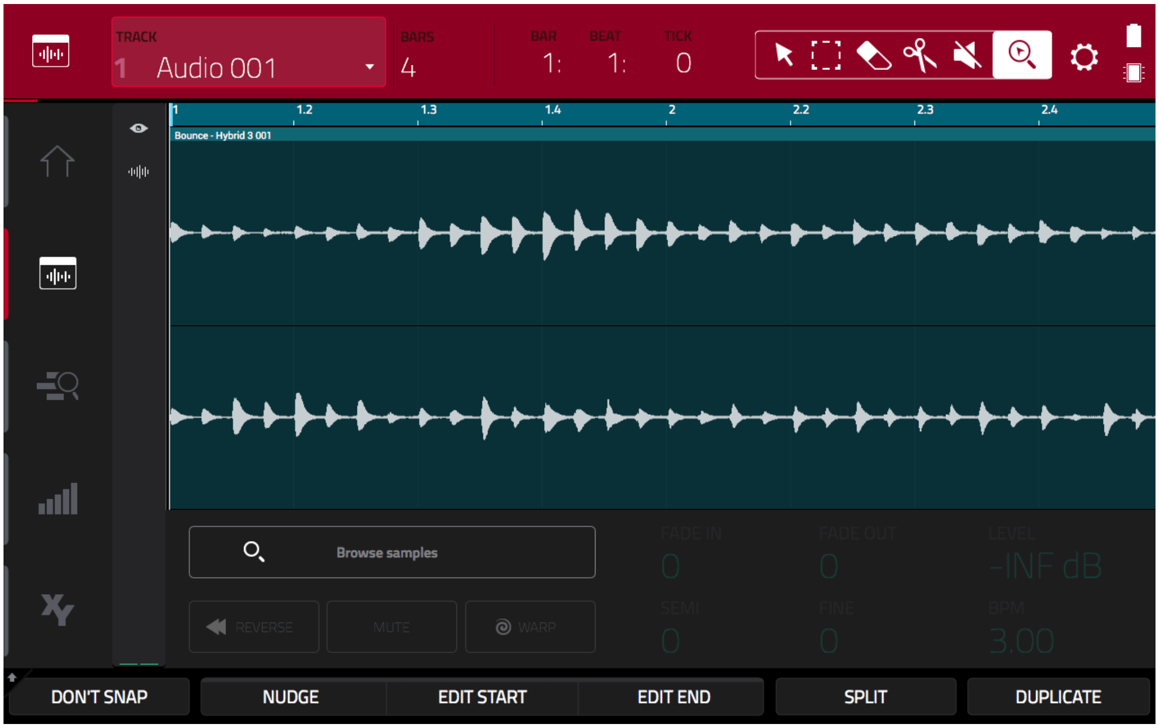
On the left side of the MAIN screen you might have noticed that the GRID EDIT icon has disappeared. This is because GRID EDIT is only used for MIDI events. Instead we have the **AUDIO EDIT** icon – hit this to enter the **AUDIO EDIT** screen (you can also access this via **MENU > AUDIO EDIT**).



Tap on **Browse Samples** to bring up the **ADD SAMPLE AT PLAYHEAD POSITION** screen:



Select the **Bounce – Hybrid** recording. Now, we want to position this recording at the very start of the audio track, so tap on the **POSITION** field and turn your data wheel anticlockwise until you see **POSITION: 1: 1 :0** . Hit **DO IT**.

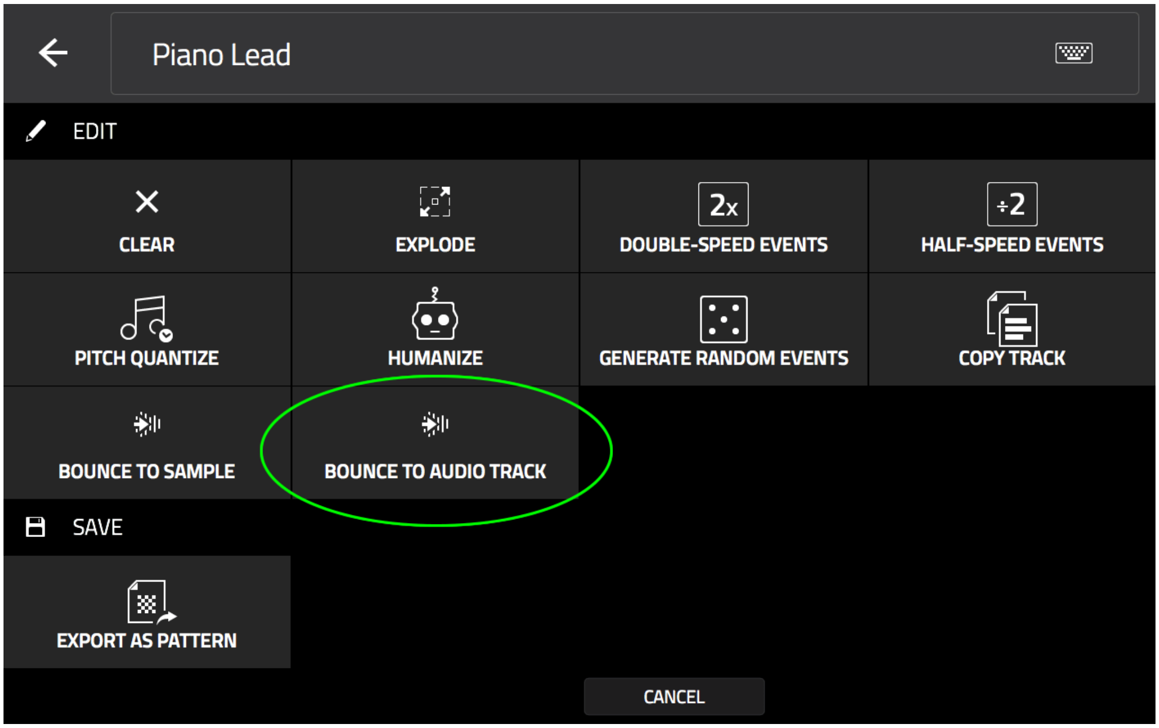


Hit **PLAY START** and you should hear the audio track play along with your existing MIDI tracks. My version is in **sequence 2** (**Demo Beat – Audio Track**).

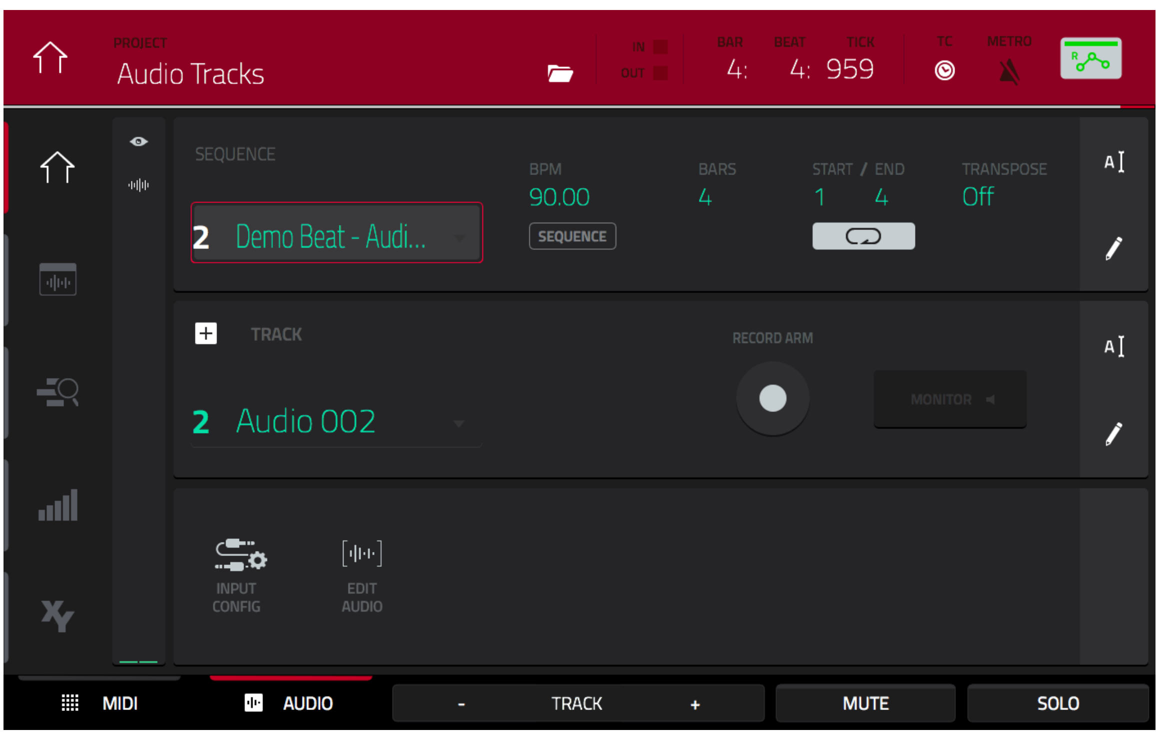
Converting Any MIDI Track to an Audio Track

In the previous chapter we bounced a plugin track to an audio sample. On the **MAIN > EDIT** screen there was also an additional option – **BOUNCE TO AUDIO TRACK**. Now that we know a bit more about audio tracks we can look at this option in more detail. However this time we're going to see how it can be used on *any* type of MIDI track – including DRUM and KEYGROUP type tracks.

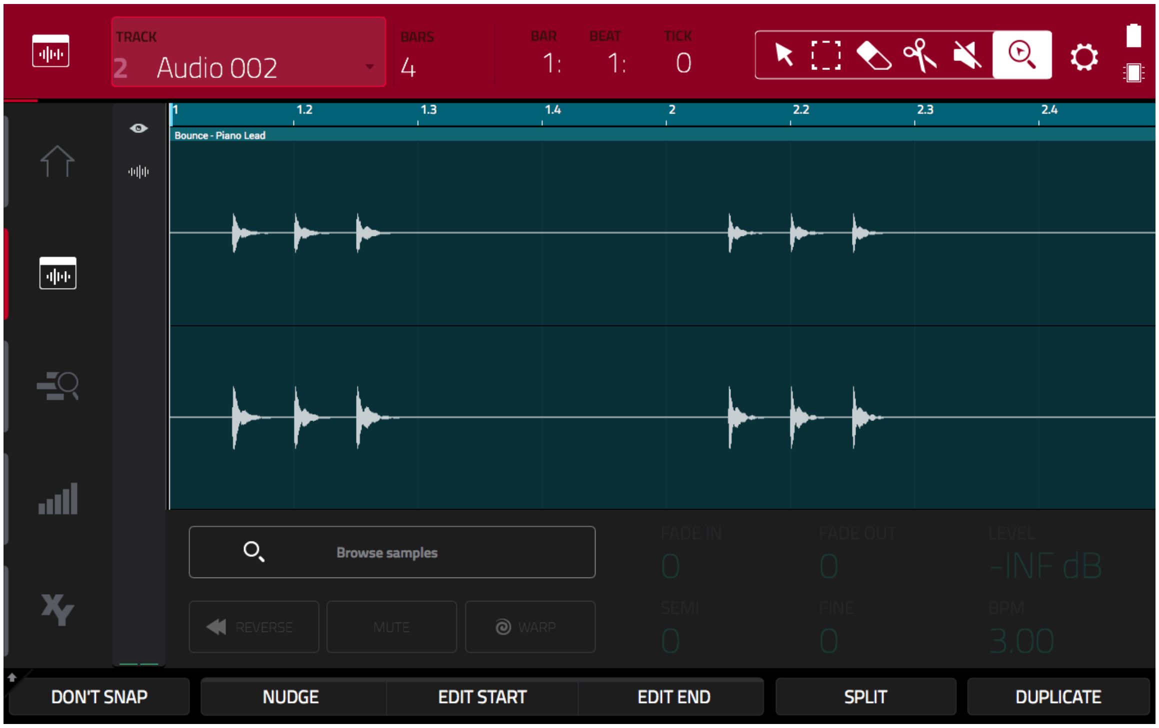
In sequence 2, go to **MAIN,** choose the **MIDI** tab at the bottom of the screen and select the **Piano Lead** track (**4**) – this is the little piano lead riff created with the **Piano Program** keygroup program at the end of chapter 11. Tap on the **pencil icon** at the end of the **TRACK** row.



Tap on **BOUNCE TO AUDIO TRACK** and after bouncing you'll be taken back to **MAIN**, where you'll see the MPC has created a new audio track:



Tap on **EDIT AUDIO** and you'll see the bounced piano lead recording:

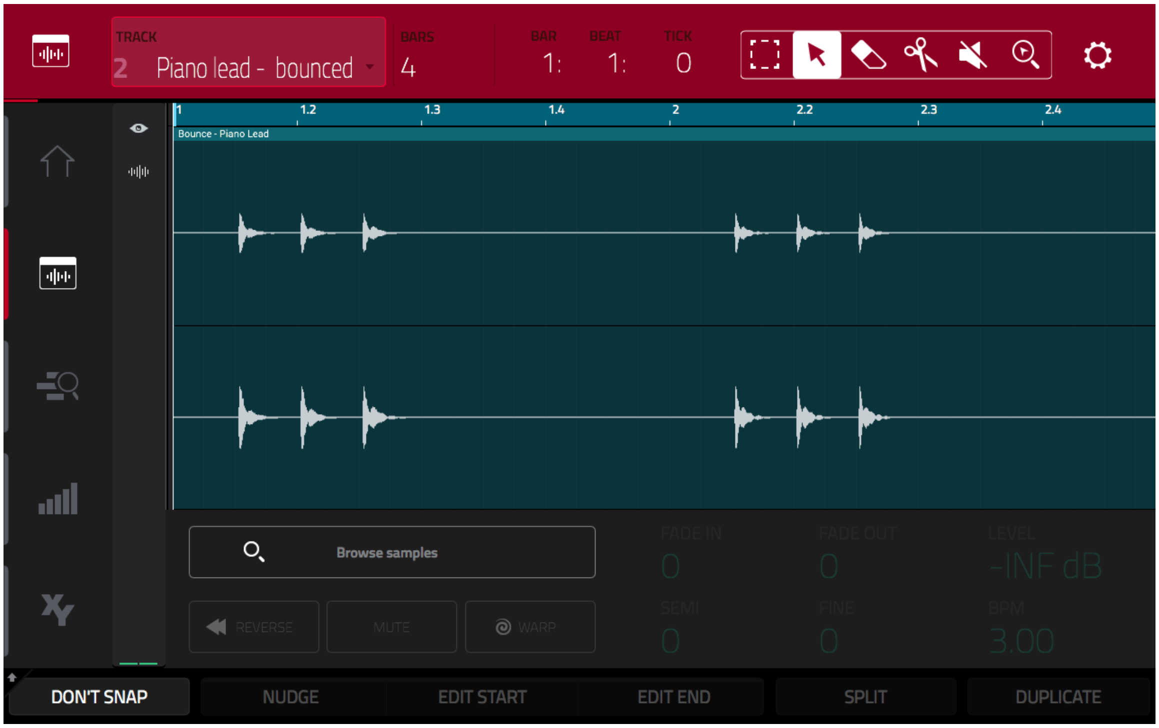


Now head back to **MAIN** and tap on **MIDI** where you'll be taken back to **track 4**. You should see that the MPC has '**muted**' this track automatically as part of the bounce process. Hit **PLAY START** and you'll still hear the piano lead play, but this time it is coming from our second audio track rather than from the MIDI track triggering the piano program.

Load up the project file **Multiple Audio Tracks.xpj** and play **sequence 1** to hear my version (I've renamed the audio tracks to indicate their sources).

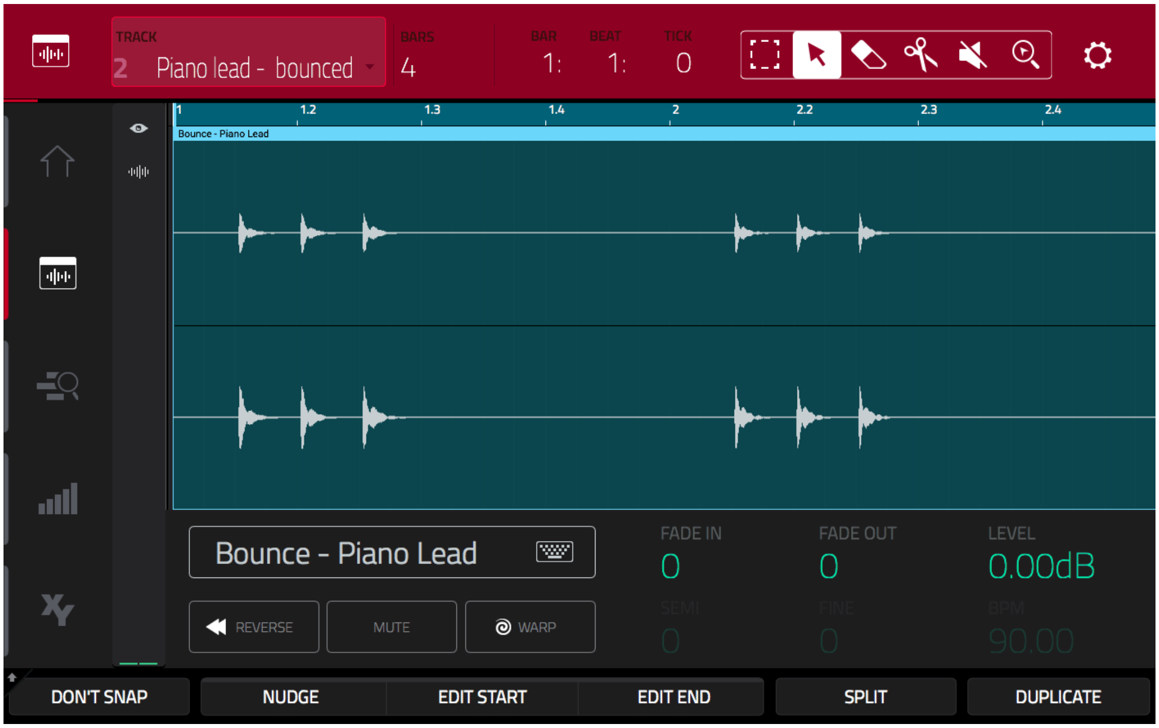
Editing Audio Tracks

In **MAIN > AUDIO** select **audio track 2**; **Piano Lead - Bounced** and choose **EDIT AUDIO:**

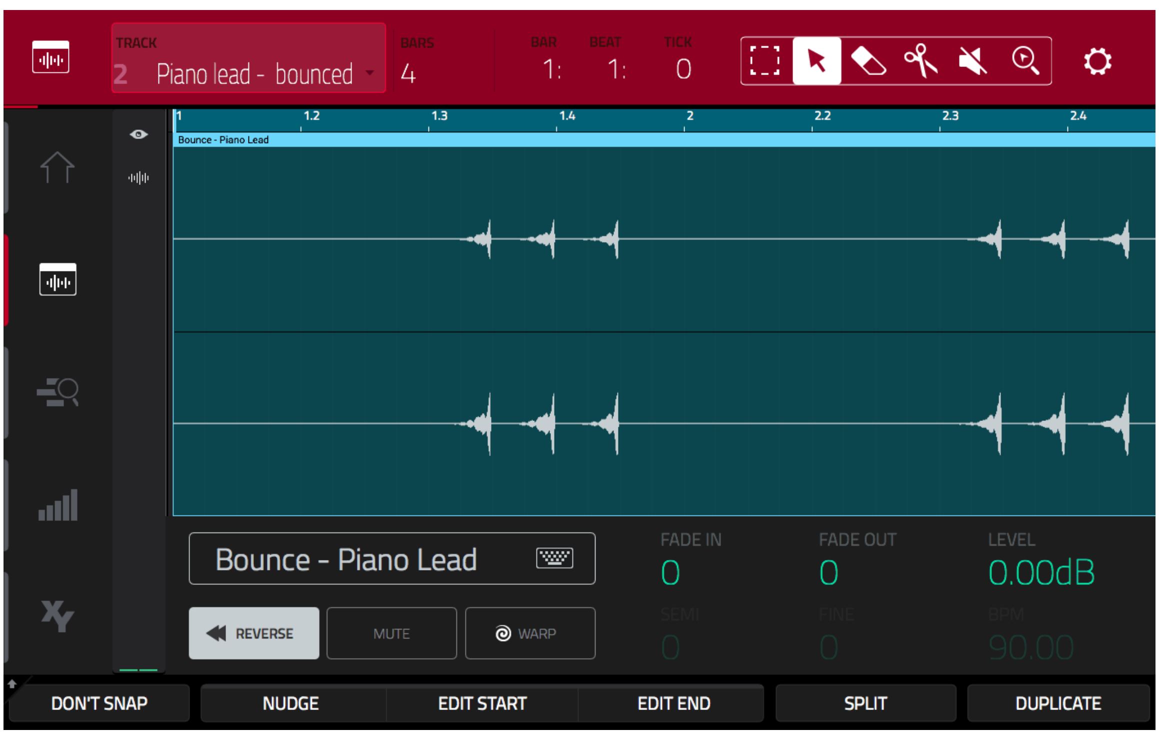


To begin editing the audio directly within the audio track environment, make sure the **SELECT** tool is selected:

Now tap on the waveform to activate it:



Let's perform a nice and simple bit of processing to our audio track – hit the **REVERSE** button and you'll see the waveform is instantly reversed:



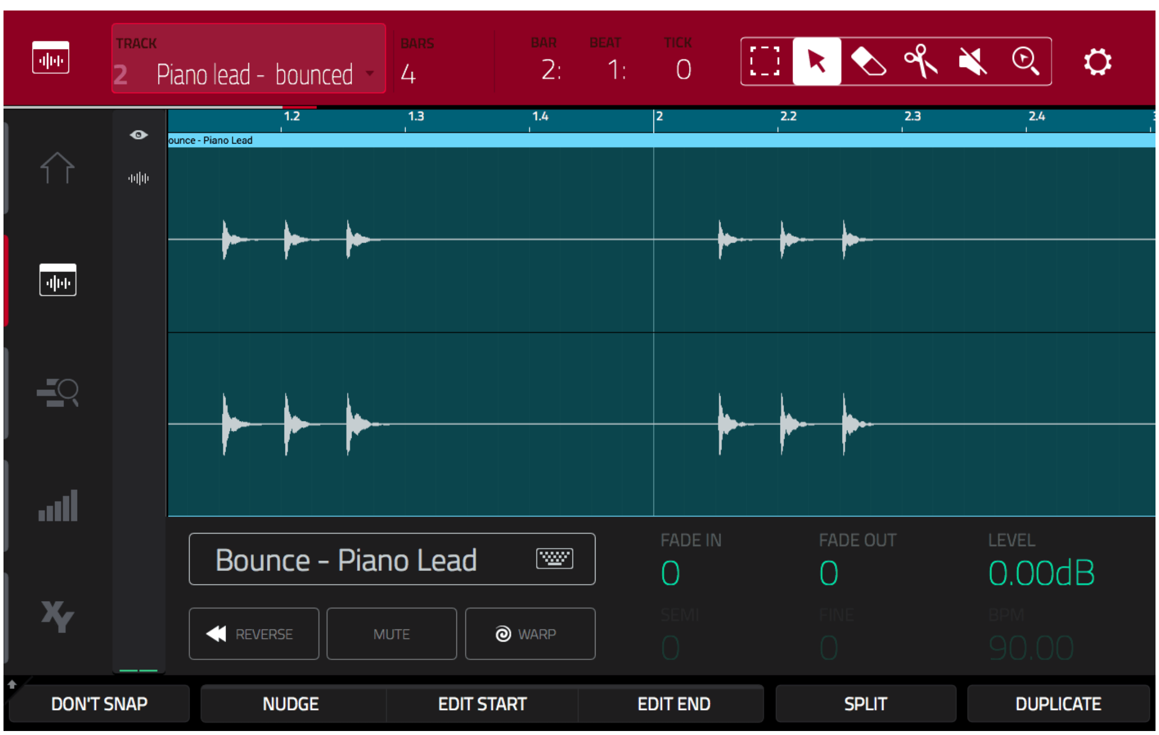
Hit **PLAY START** and you'll hear our piano lead sound playing backwards. Hopefully this has already shown you one of the creative advantages of audio tracks – when you were working in the MIDI environment, it would not have been possible to 'reverse' your track like this. In fact you probably would never have even considered the idea of reversing the lead line in the first place.

Select **sequence 2**. Here I placed a copy of the original sequence; go to **EDIT AUDIO** and hit **PLAY START**. As you can see and hear, in this sequence the audio continues to play in a forward direction, yet it's using the same audio sample.

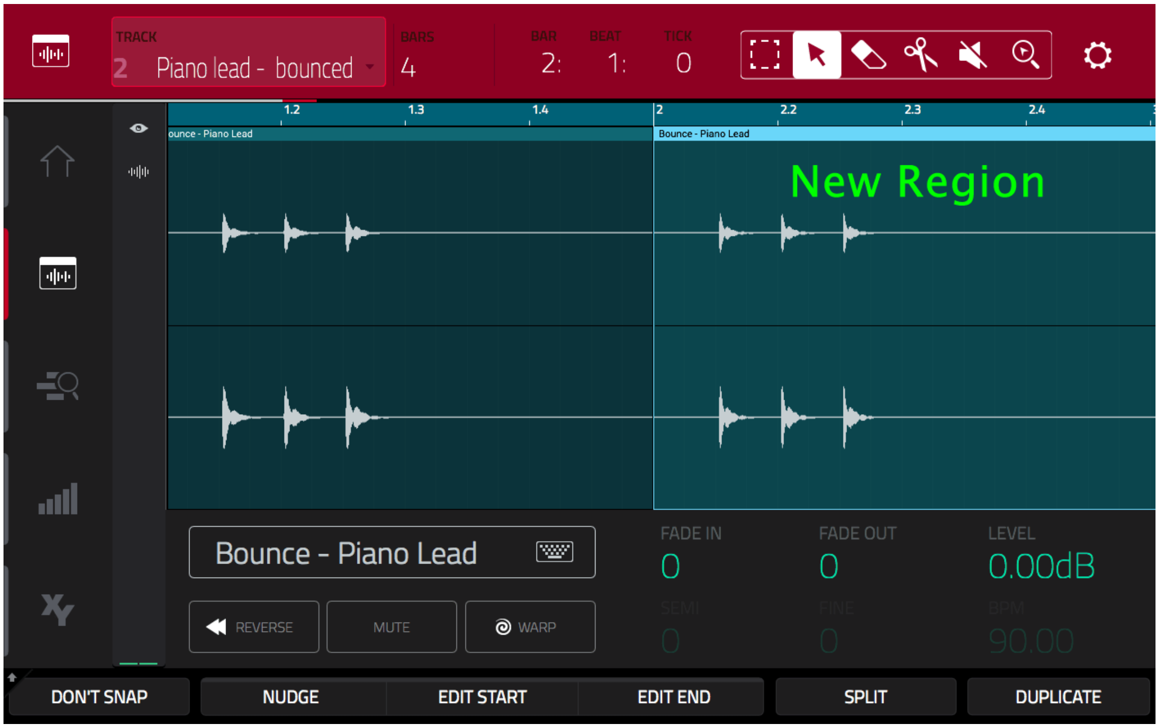
Within audio tracks you are able to use *instances* of the same sample/recording in multiple audio tracks across multiple sequences and happily edit each instance independently; you can also edit the 'master' recordings (in this case **Bounce – Piano Lead**) in SAMPLE EDIT, however any edits you perform there will affect all instances across all audio tracks that use this sample (which is the same concept as when using instances of the same sample across multiple drum pads and programs).

Splitting Audio Tracks

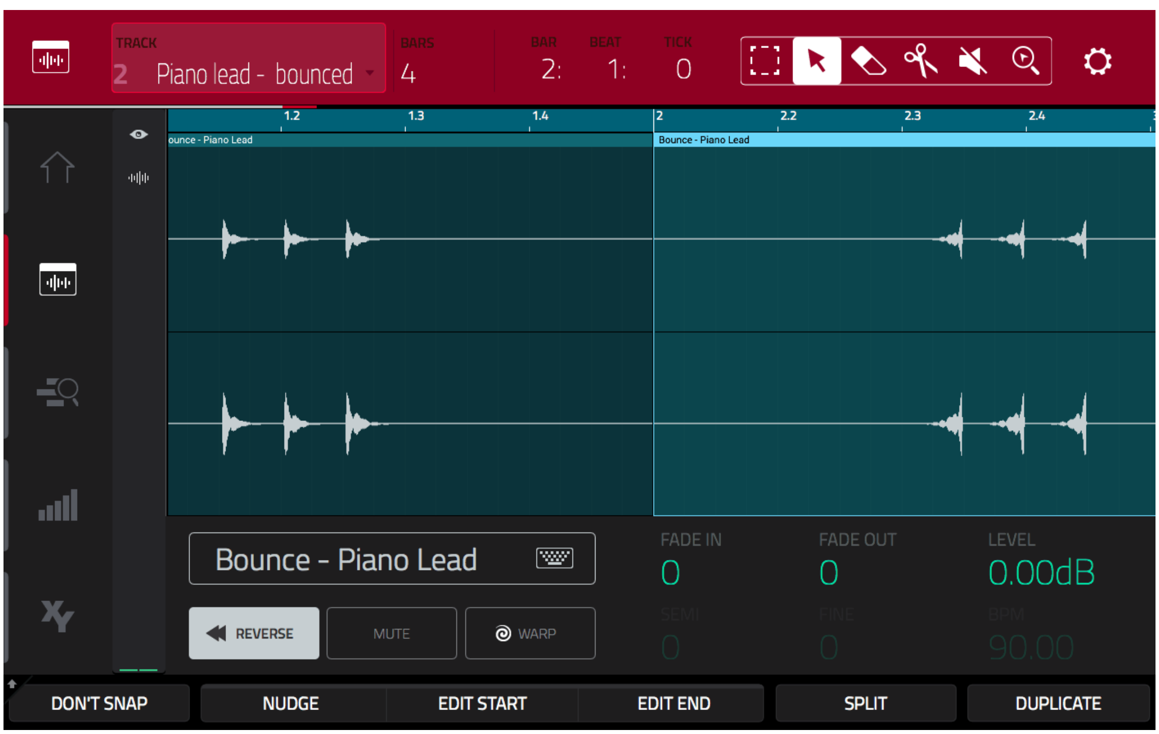
We can also split audio tracks into segments and then process each segment differently. Staying in **sequence 2, track 2,** tap on the time locator at the top of the screen and turn the data wheel so the time locator reads **2: 1: 0** (the start of the second bar):



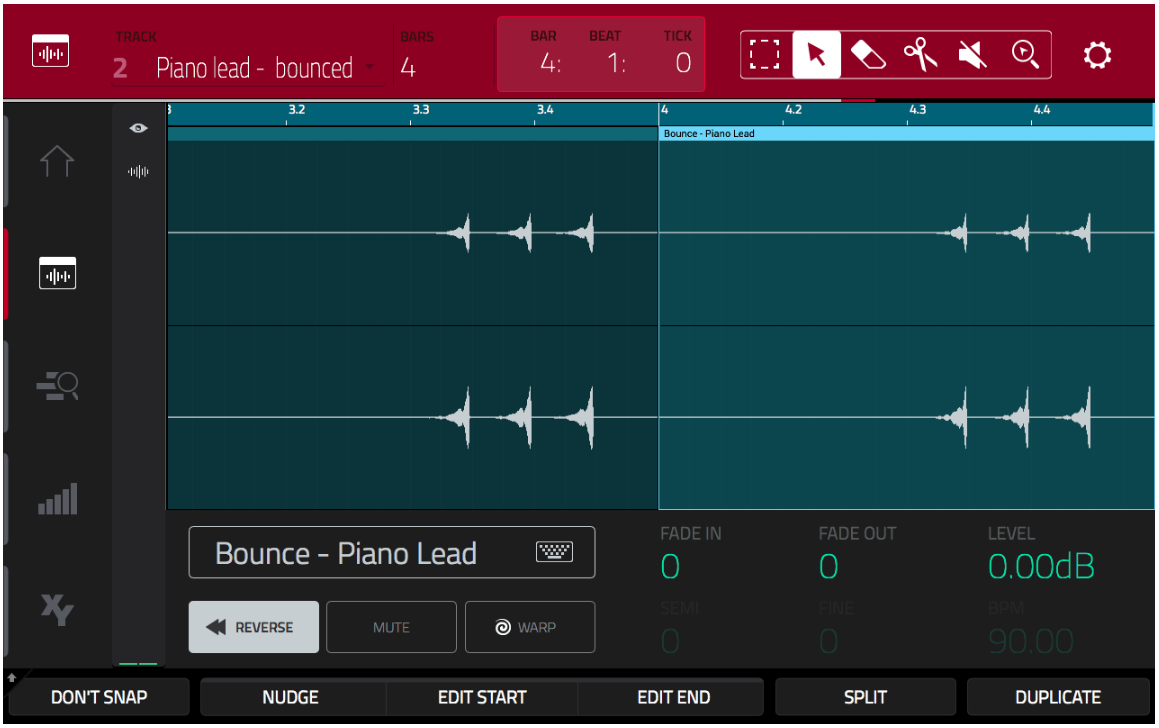
Tap on the **SPLIT** button – this creates a new region at the selected time:



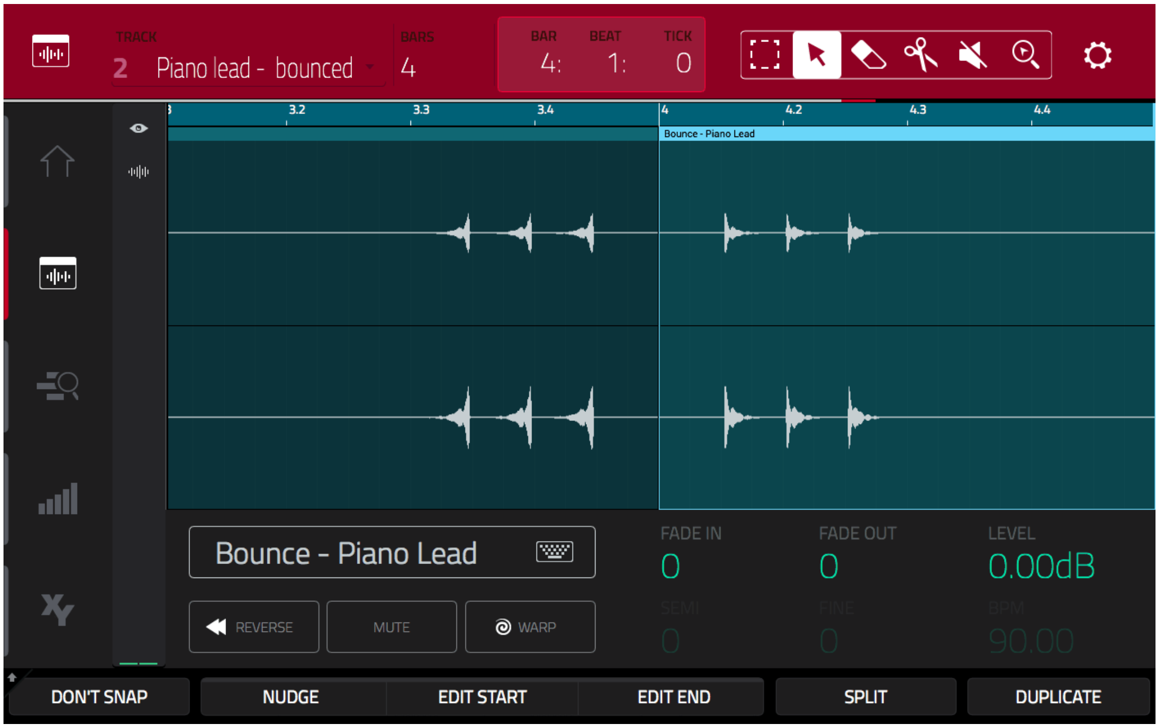
Tap on this new region to select only this region. Hit the **REVERSE** button to reverse this second region.



Hit **PLAY START** to hear how this sounds – as you can hear, the rest of the audio track is now played in reverse. Move the time locator to **4: 1: 0** and press **SPLIT** again.



Ensure the new region is selected and press **REVERSE** to ;de select' the reverse function. This split portion now plays in a forward direction.



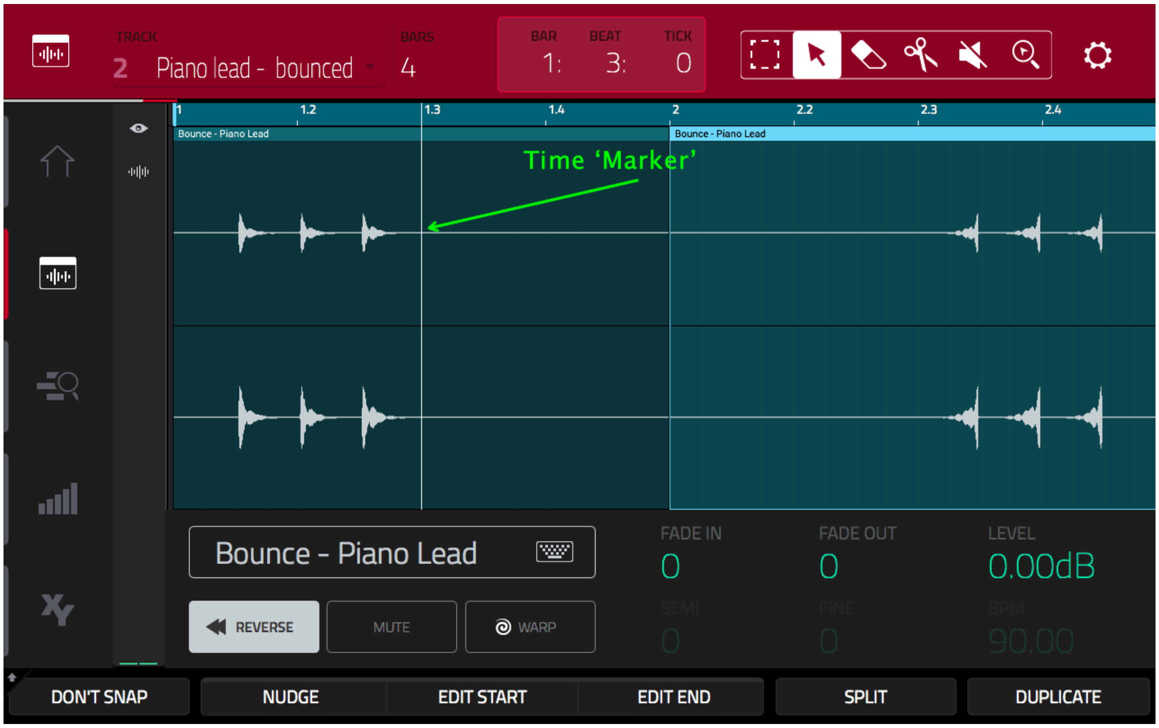
You can play my version via **sequence 3**.

Moving Regions

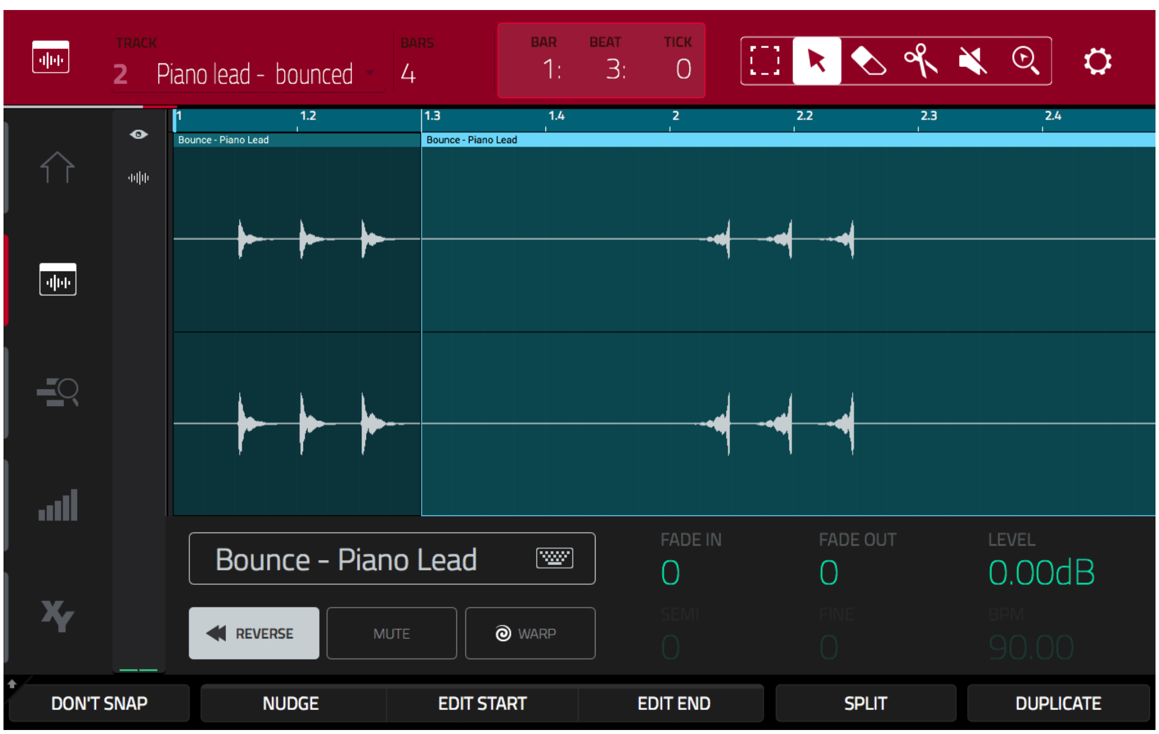
I'd prefer it if the reversed regions still played at the same relative position within each bar, as currently those reversed portions play quite late in the bar. This is very easy to do.

First, let's set a time location 'guide marker' – I just want to place the cue playhead at a specific position so it's easier for you to move the region to the exact spot I want you to. Hold down **NOTE REPEAT** and select a **1/16 TC.**

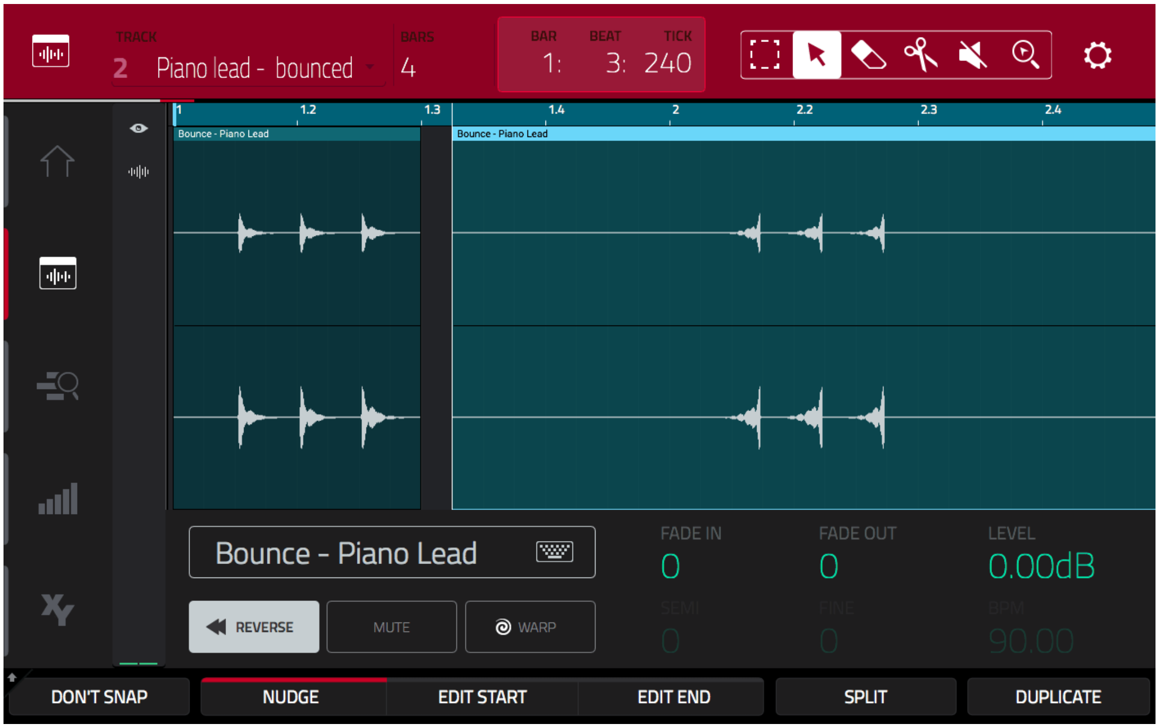
Tap on the time locator at the top of the screen and turn your data wheel to **1: 3: 0**.



Now tap and hold the reversed region (which is currently place at the start of bar 2) and 'drag' it to the left until you reach the cue playhead. Release the region.



Hit **PLAY START**. Sounds better, but we can continue tweaking the position. Tap on the **NUDGE** button - this allows us to move the region in controlled increments (based on the current TC, which is why I asked you to set this previously); turn the data wheel clockwise and nudge the region one 'click' to **1: 3: 240:**



Press **PLAY START** to hear how this sounds. My version is on **sequence 4**.

I'll introduce some more advanced audio track editing features in chapter 25!

013 Using The MPC Mixer

Our Demo Beat is starting to become increasingly complex with a decent array of MIDI and audio tracks. You can probably hear that the overall stereo 'mix' of the entire sequence is a bit untamed at present and could do with some attention. Let's look at how we can handle these multiple tracks in a more efficient and effective way.

MPC Mixing Options: An Overview

An important part of any sequence is the eventual need to ensure that all the various sounds are nicely mixed together, which at the most basic level means ensuring that their relative volumes are well matched (i.e. nothing too loud or too quiet), but as you gain more confidence in your production abilities will eventually encompass a wide range of skills such creating an effective 'stereo field' (from careful use of panning, eq and volume adjustment), as well as judicious use of effects, dynamics and filtering to fix and tame any problems or purely for creative effect.

The MPC software has a massive range of mixing options built into it and it can get quite confusing, so let's have a quick look at the main mixing options and their primary uses.

The Pad Mixer

We already met the pad mixer in chapter 007 when we used it to adjust the 'mix' of our drum kit. The pad mixer is perfect for this kind of 'in-program mixing' where you have multiple pads or keygroups within the same program that need to be adjusted *in relation to each other*, be it relative volumes, panning or effects.

The pad mixer is however much less suited to performing 'global' mixer adjustments to an *entire* program, for example applying an effect across the whole program or changing the output volume of the entire program.

Program Mixing

Program mixing allows you to adjust mixing parameters across an *entire program*. For example, the program mixer channel could reduce the overall output volume of the Piano Program, it could be used to apply a reverb effect across the entire program, and it could also be used to route the entire program through a different set of audio outputs.

Track Mixing

Track mixing is used to adjust mixing parameters on a *single track*, be it a MIDI or audio track. MIDI tracks can have their volume and panning adjusted, while audio tracks can also have effects applied. Any changes made via the track mixer will only affect that one track. If you have other tracks that use the same program, those tracks will be unaffected.

Master Mixing

This allows the global tweaking of the entire output of your mix; for example addition of a compressor or EQ across the stereo '1/2' output.

What Happened To Submixing?

Submixing allows us to 'group' together selected pads and keygroups from one or more programs so we can perform mixing operations on all these sounds simultaneously. For example, all the individual pads from a drum kit could be grouped together in the same submix to have a compressor applied across the entire kit.

For reasons unknown, Akai has completely removed the option of 'submixing' from standalone mode. It's still there in controller mode, but I'll work around its 'standalone' omission as we progress through the book.

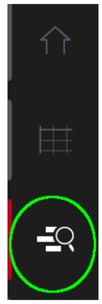
Let's go through some practical examples to help explain the key differences in these various mixing options.

Track View Mode

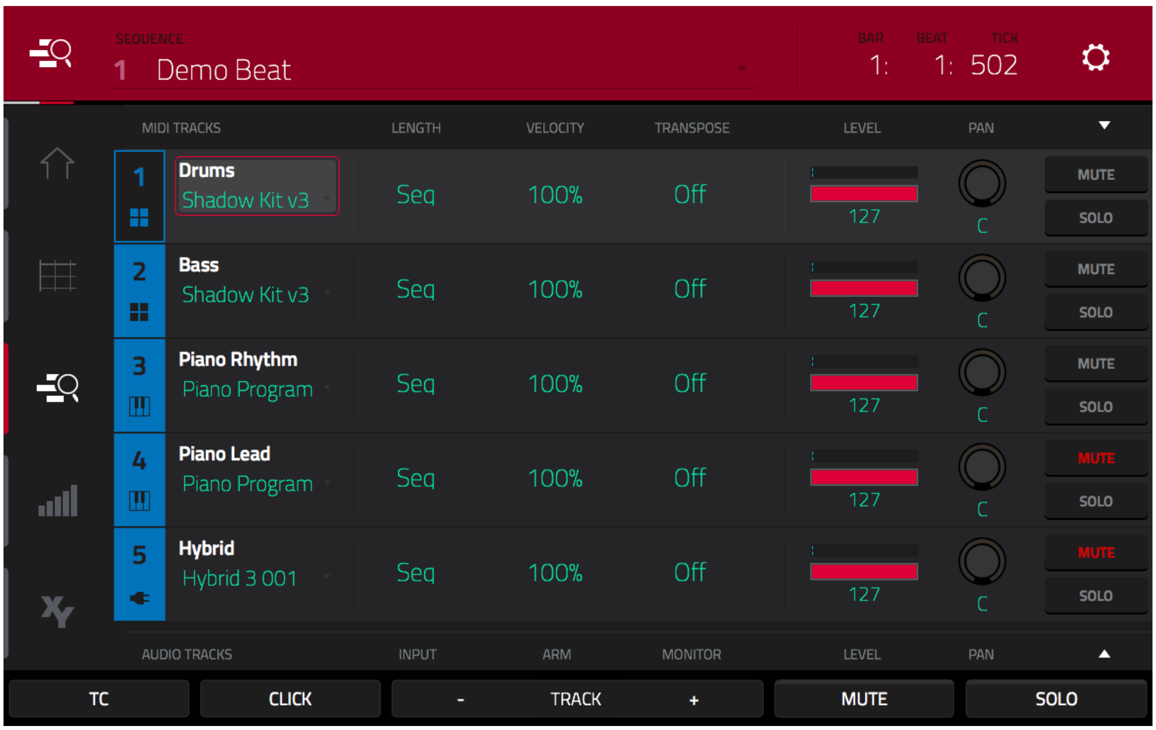
So far when dealing with tracks in MAIN we've been limited to viewing a single track at a time, which is fine when you are 100% focussed on that single track but as your beat begins to take shape you might want to see an overview of all the different tracks in a particular sequence.

Load up the project file, **Mixing.xpj**. This will load up our current 'Demo Beat' sequence to sequence 1 of our project.

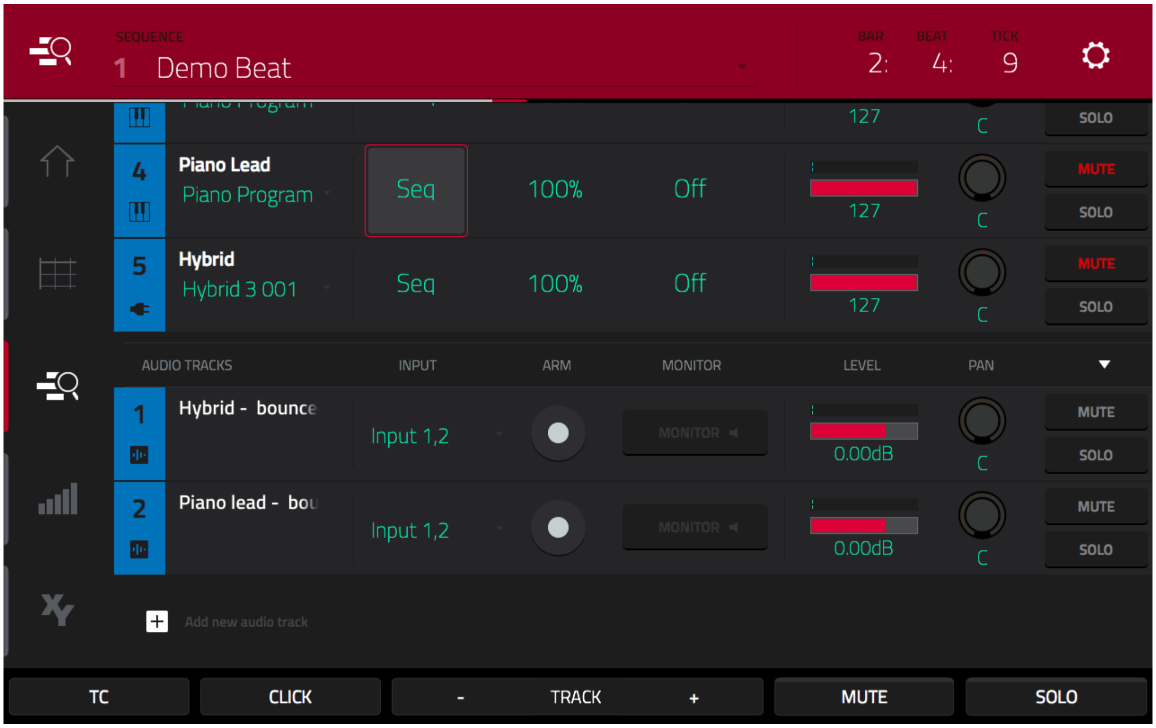
From within **MAIN**, hit the **TRACK VIEW** icon at the left of the screen.



Alternatively, go to **MENU** > **TRACK VIEW** or quickly double tap **MAIN**:

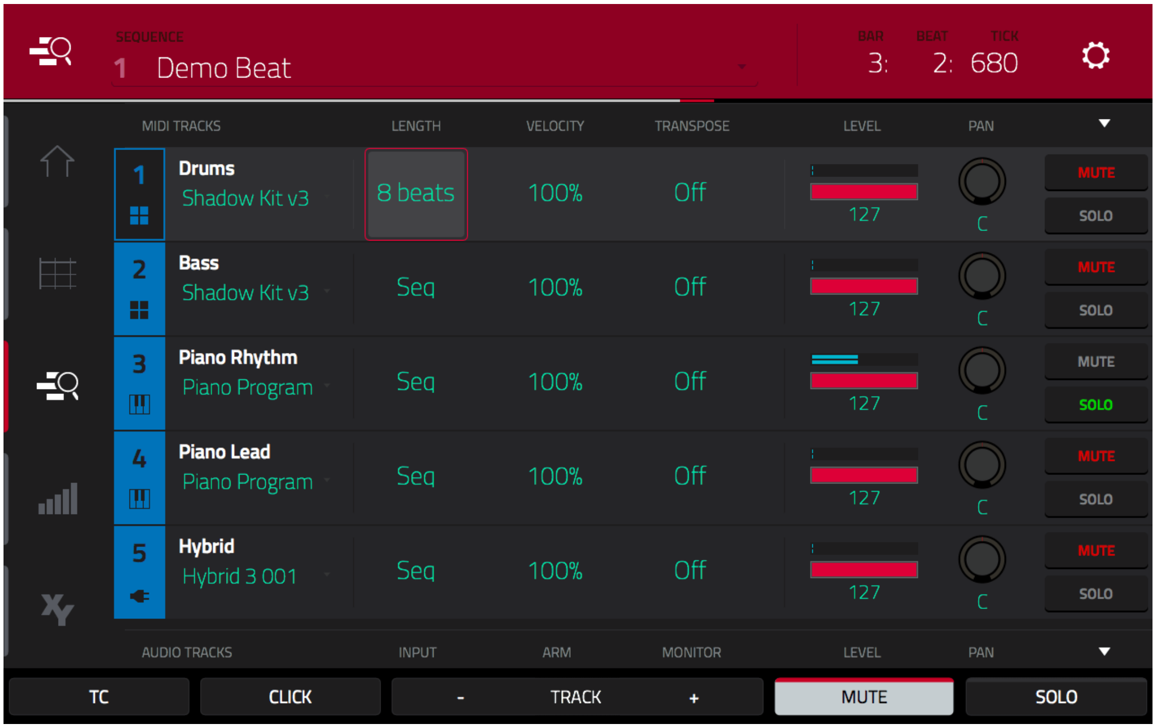


Here you can see how TRACK VIEW shows all five of the MIDI tracks in our current '**Demo Beat**' sequence. Hold and drag the screen upwards with your finger and you'll see all the audio tracks in your sequence:



A red **MUTE** button at the end of a track row lets you quickly see that the Piano Lead and Hybrid tracks are already ‘muted’ (remember we have bounced those tracks down to audio tracks) – you can however unmute any muted track by hitting the red MUTE button.

Tap on the **SOLO** button for the **Piano Rhythm** track.

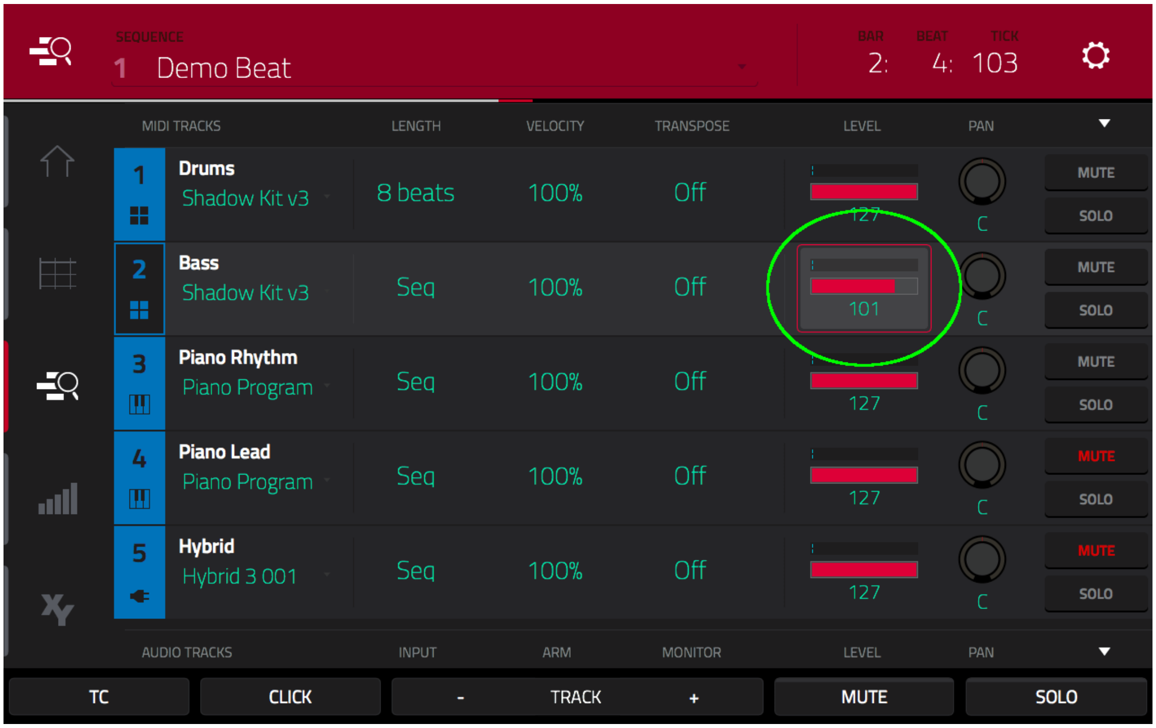


With a track solo'd all other tracks in your sequence become muted. This allows you to carefully listen to that track without any distractions. Hit the **SOLO** button again to return the track to its normal state.

Adjusting Levels in Track View

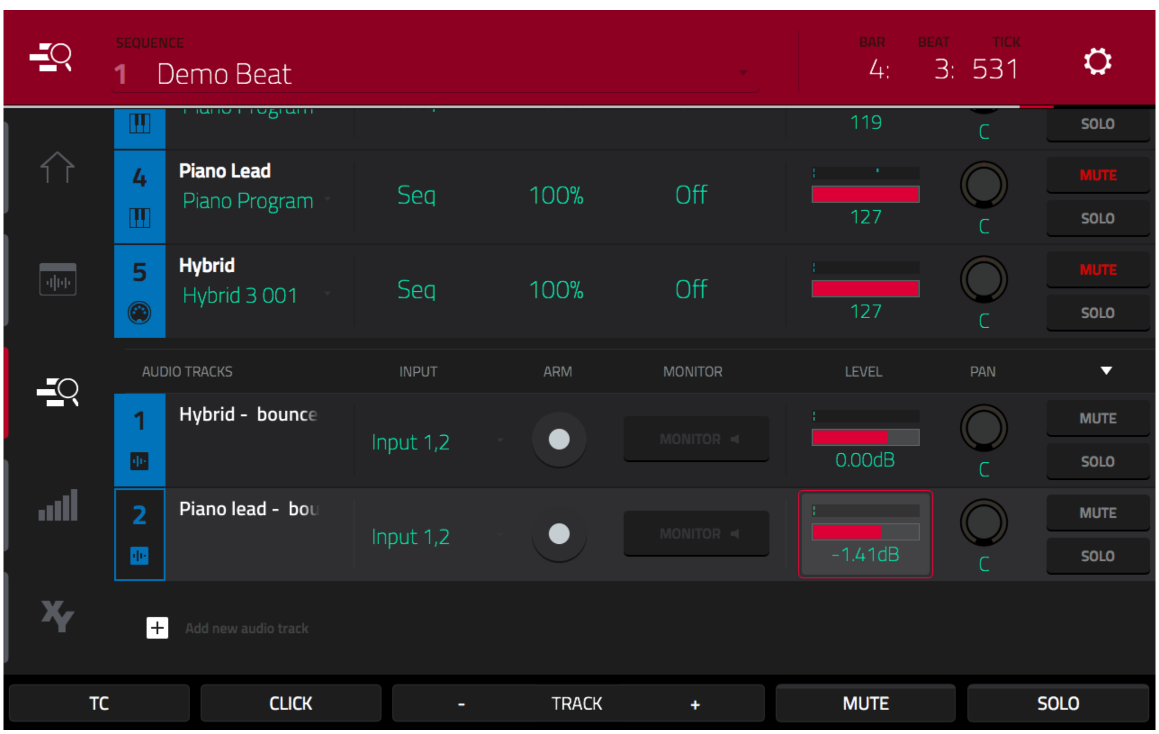
Mixing options in track view are relatively limited, but if all you want to do is tinker with output volumes of your tracks, then track view can be quite convenient. With your sequence playing, tap on the **LEVEL** parameter for the **Bass** track and use the data wheel to reduce the output level of this track. Remember to hold down **SHIFT** while you turn so you can change the LEVEL in single units.

Set the **LEVEL** for **track 2** to **101**.



Tap on the **Piano Rhythm** track and reduce the **LEVEL** to **119**.

Now drag down to the audio tracks. Tap on the **LEVEL** for **audio track 2** (**Piano Lead**) and reduce this to around **-1.41dB**.



LEVEL: MIDI vs Audio Track

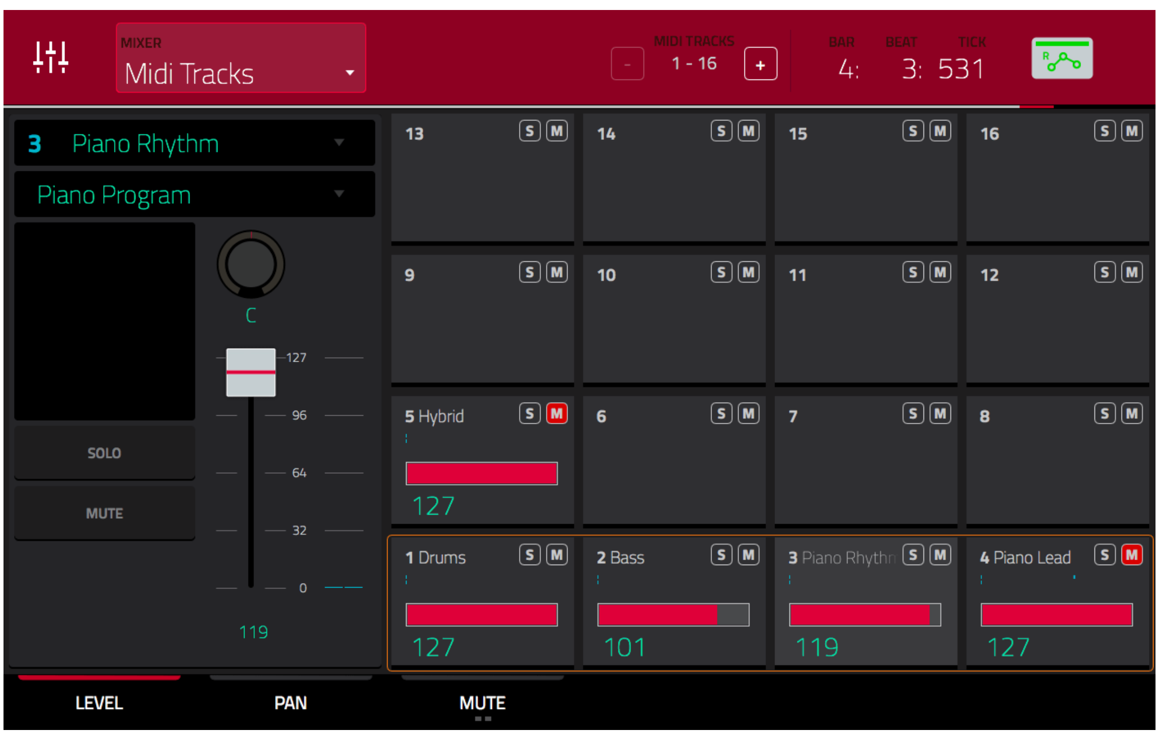
Notice how audio tracks have their levels measured in decibels (dB) while MIDI tracks have no units and a range of 1 to 127? This is because 'MIDI' is not audio, a MIDI event on its own does not have any actual audible 'output', it is the sound source that is triggered that has the 'audible' output that can be measured in decibels.

This is just a quick rough level adjustment and as we add effects, eq, change panning and so on, you'll need to continue making small adjustments to levels to compensate.

TRACK VIEW is great for a general visual overview of our tracks, but it is not the only place where we can adjust track volumes, panning and mutes.

The CHANNEL MIXER

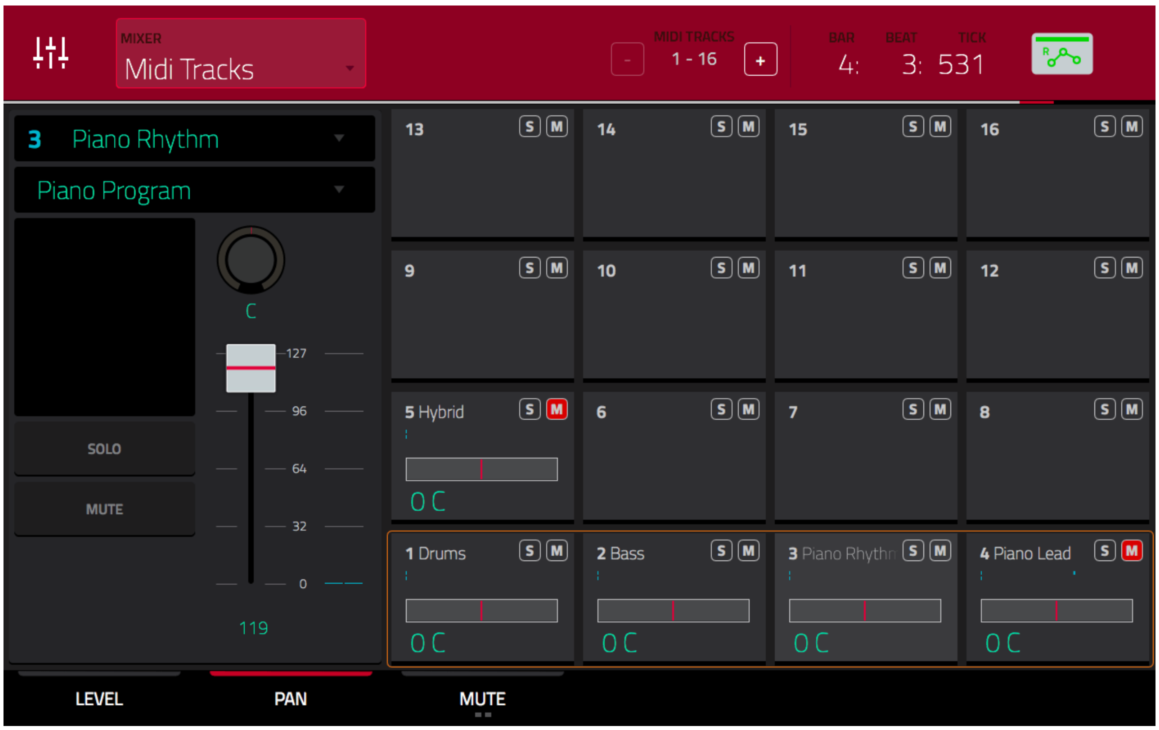
Go to **MENU > CHANNEL MIXER** (or use the dedicated **CH MIXER** button on the MPC X). In the top left of the screen, select **MIXER: Midi Tracks**.



The **MIDI track** channel mixer offers a more traditional looking mixer on the left side of the screen with a full LEVEL slider and PAN dial, while offering a view of levels and mute/solo status for up to 16 tracks on the right hand side.

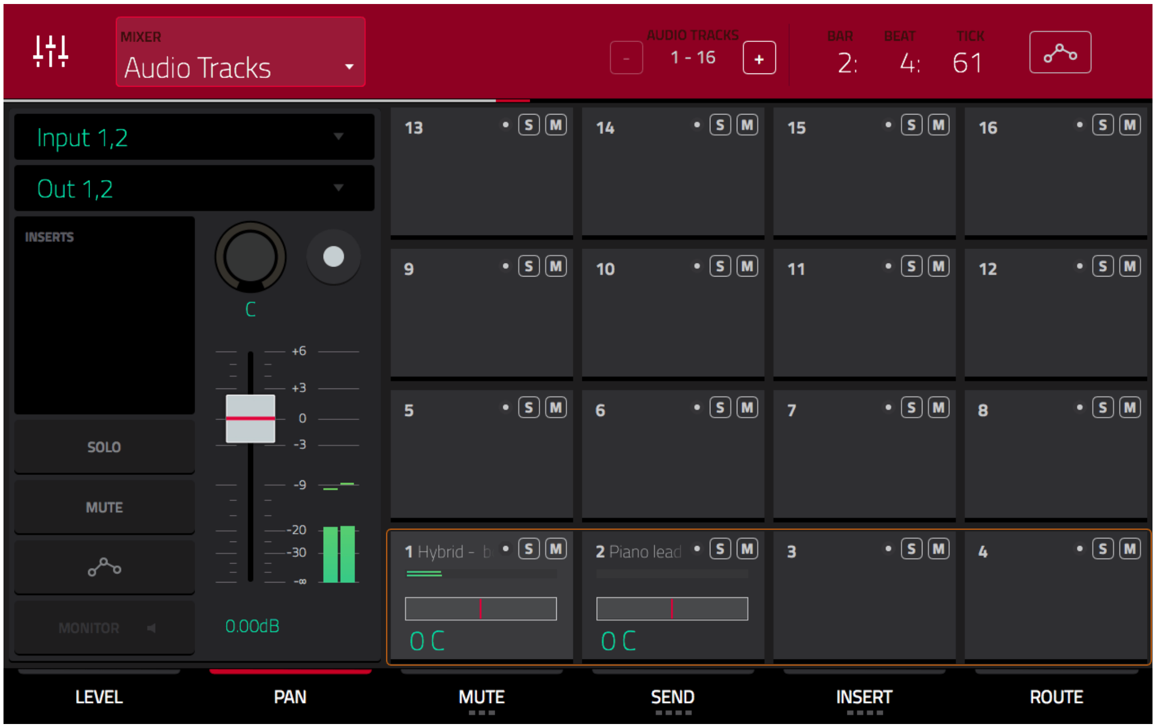
Tap on one of the tracks on the right hand side of the screen and the left side reflects the current mixer settings for that particular track.

The default screen shows LEVELS on the right hand side, but hit **PAN** and this changes into a panning overview of all MIDI tracks.



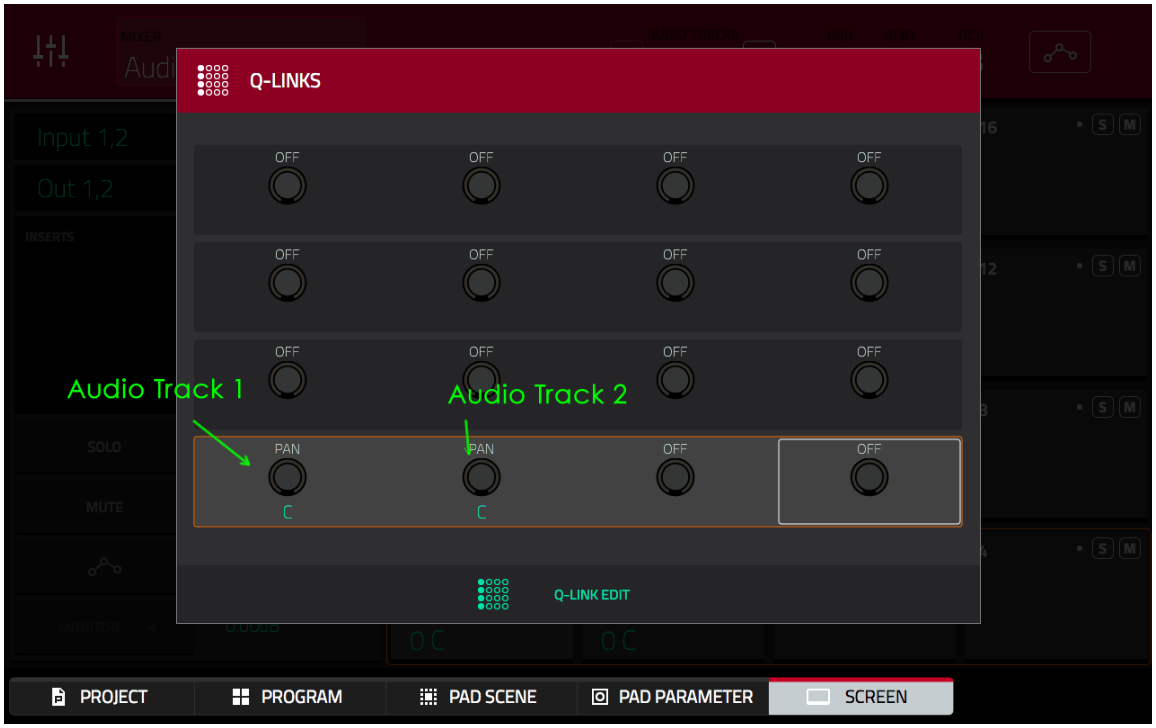
So if you are working on mixing the stereo field, this screen will most likely provide the faster workflow.

Tap the top left of the screen and select **MIXER: Audio Tracks**:



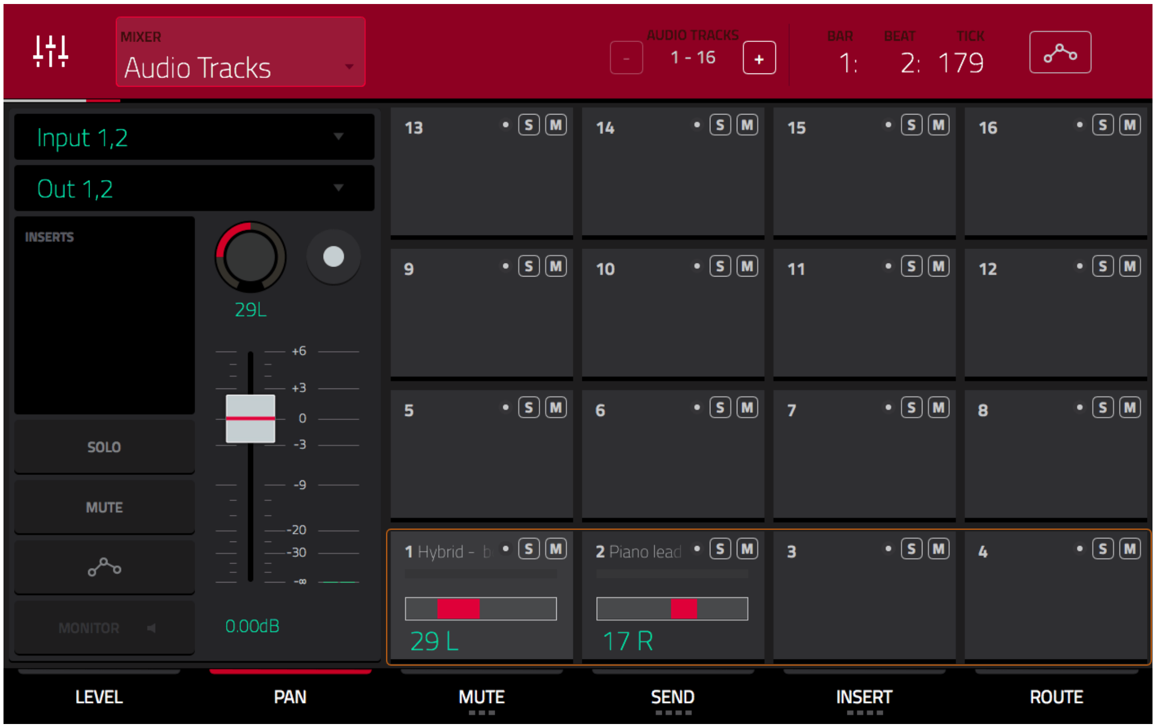
Just like the MIDI track channel mixer, the Audio Tracks channel mixer provides us with an immediate overview of all audio tracks with the traditional slider/dial option on the right and the left hand side displaying all currently used audio tracks (each track is an on screen 'pad').

Let's adjust the **panning** for both these audio tracks. There's three ways to do this; firstly you can tap on one of the tracks on the right hand side and make adjustments with the left hand side 'pan' dial. Alternatively, you can adjust the linear slider within the onscreen track selection 'pad'. My favourite method is to use the Q LINKS.



Hit **PLAY START** and start turning the **top Q-LINK dial in column 4** (**Q-LINK 16**) – this adjusts the **PAN** for **audio track 1**. Do the same for **audio track 2** using the **second Q-LINK from the top in column 4** (**Q-LINK 12**).

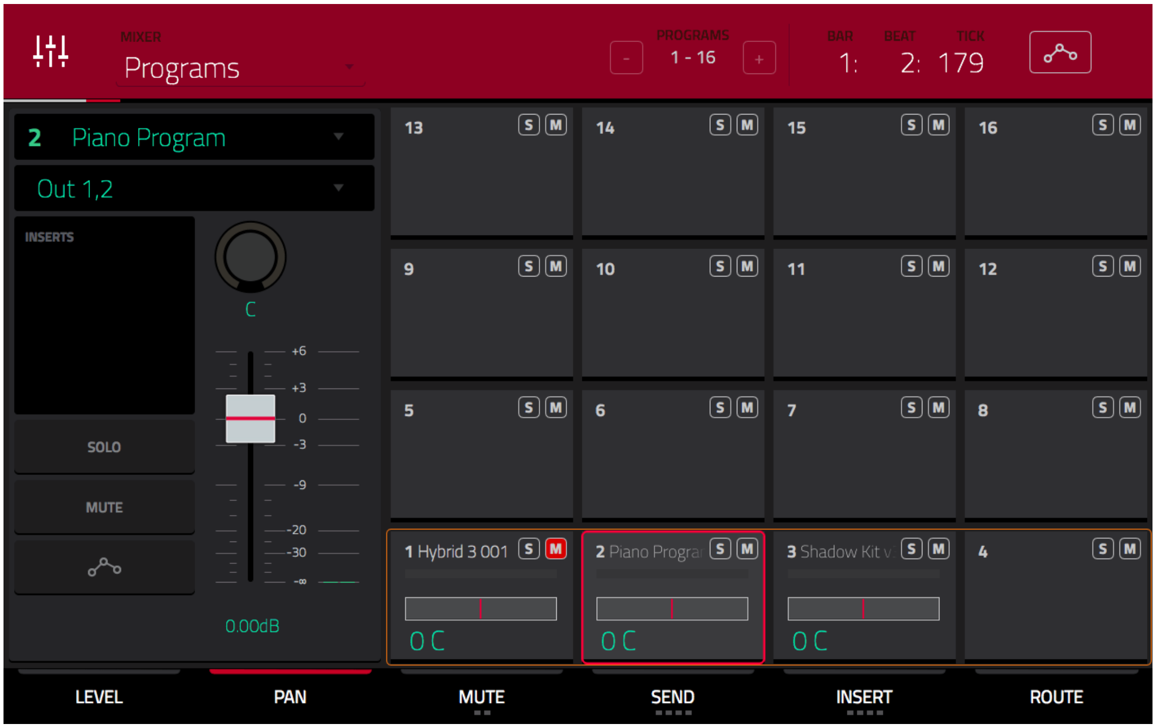
I panned track 1 fairly aggressively to the left, while track 2 was panned more subtly to the right, this has given the two melody lines as nice separation.



As with any mixing adjustments, you're obviously free to continue tweaking as your session progresses – just keep experimenting.

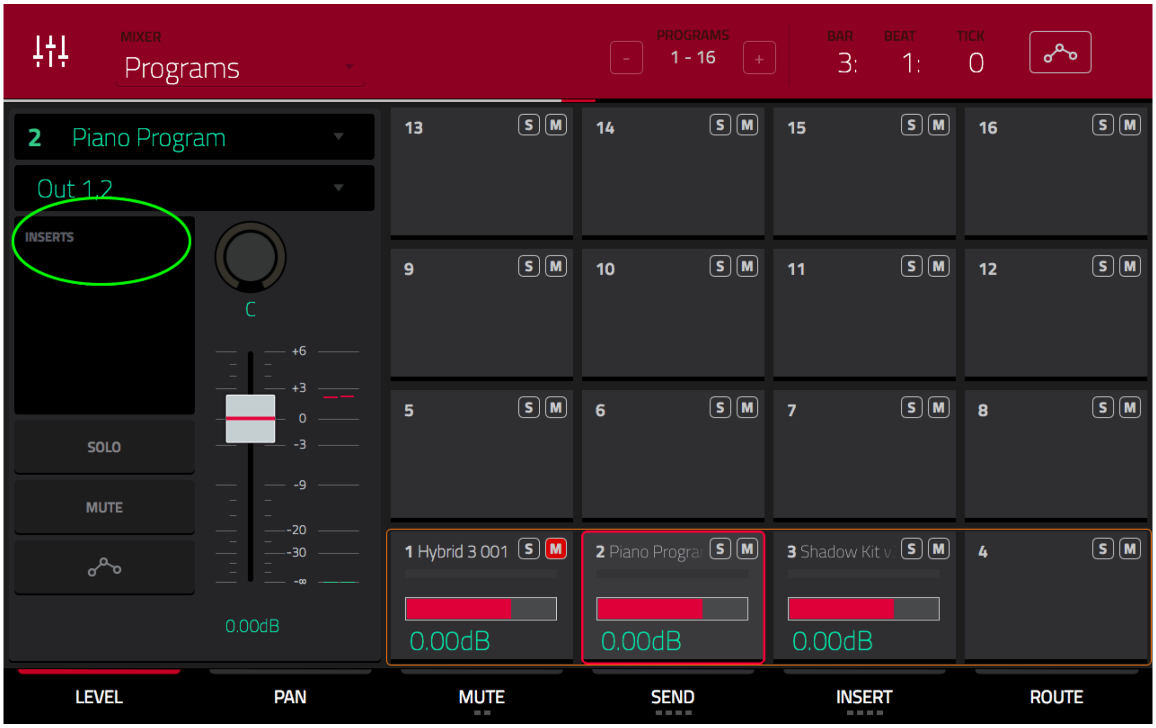
The Program Mixer Channel

From the top left of the screen, select **MIXER: Programs**:

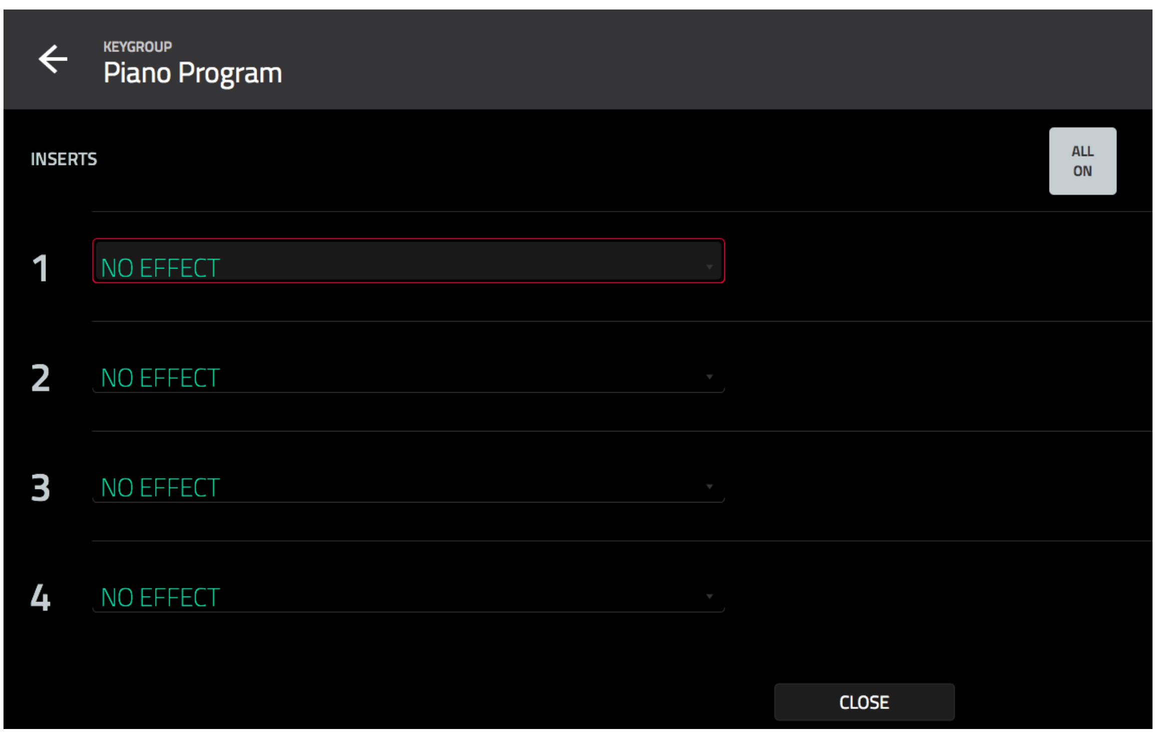


The **Programs channel mixer** shows all the active (MIDI based) programs in your sequence. As you can see, the Hybrid plugin program is there (but muted), along with your Piano keygroup program and the Shadow Kit program which contains our drum sounds and bass note.

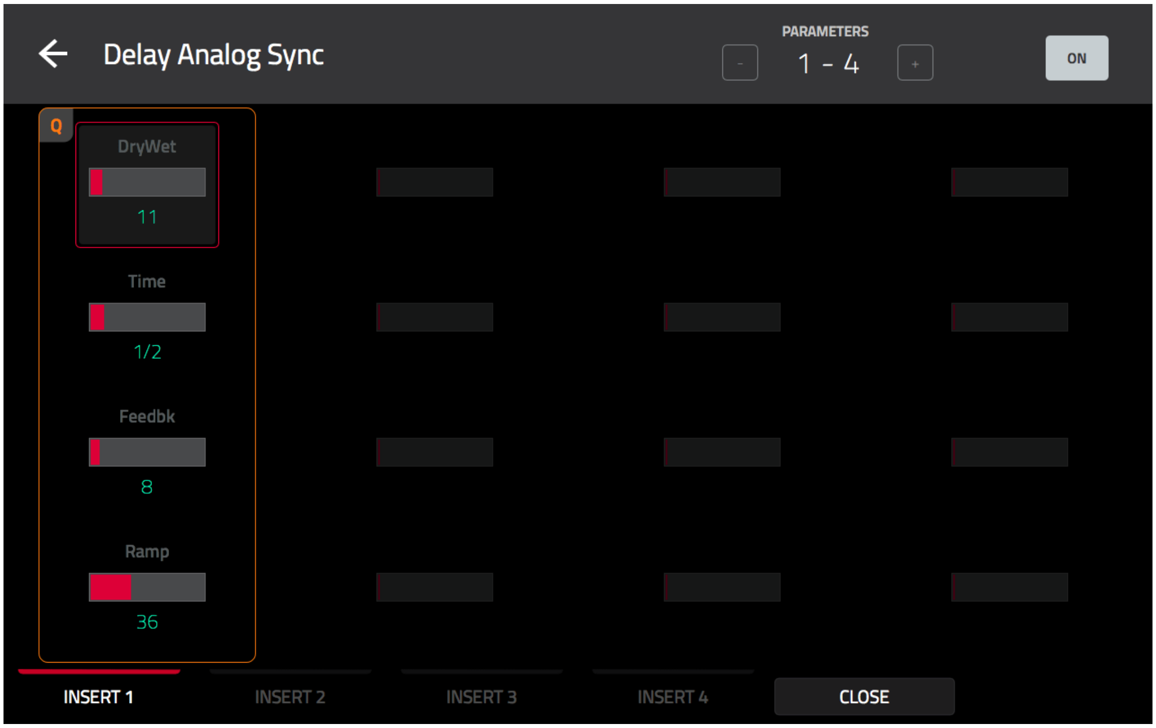
In this screen we can apply mix adjustments to an entire program. For example we might want to add some delay effect to the **Piano** program. To do this, tap on the onscreen box that represents the **Piano program** so it is now selected and tap on the ‘**INSERTS**’ box on the left hand side:



This brings up the familiar **EFFECT INSERTS** window:

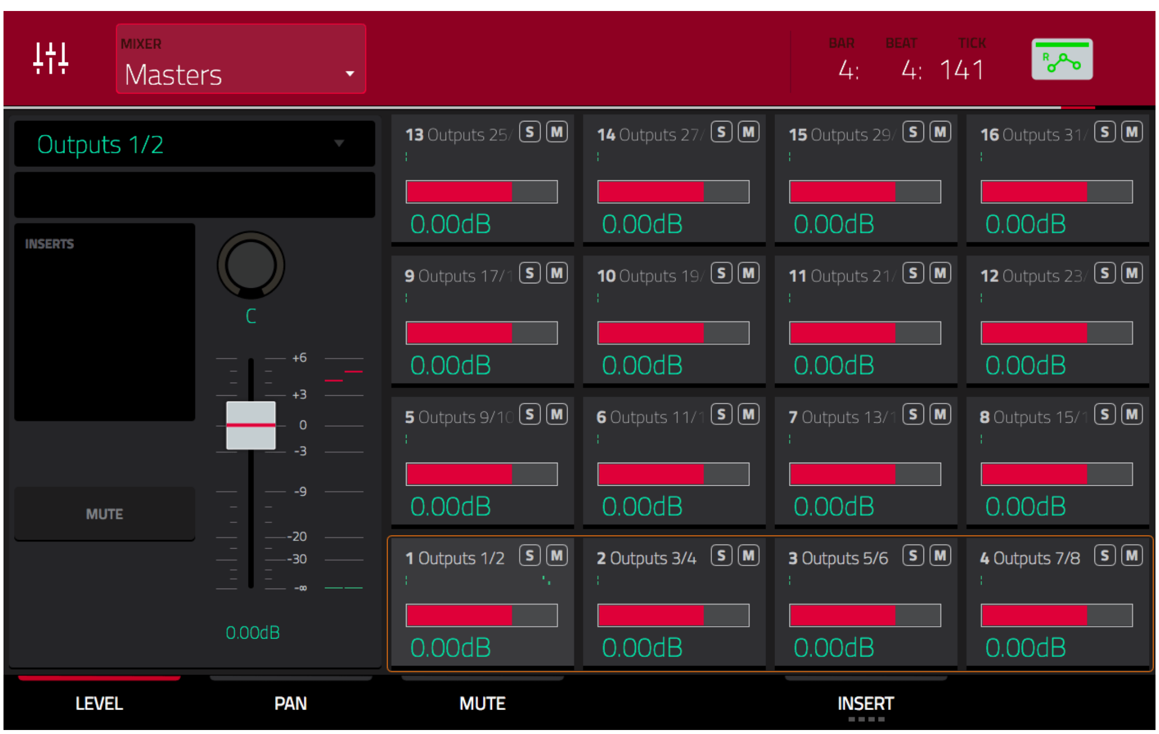


Select the **Delay Analog Sync** effect and configure it with the following parameters for a fairly subtle synced delay (**Dry/Wet:11**, **Time:1/2**, **Feedback:8**, **Ramp: 36**):



The MASTERS Channel

After your audio has passed though the various routing, it finally reaches the **MASTER channel**.



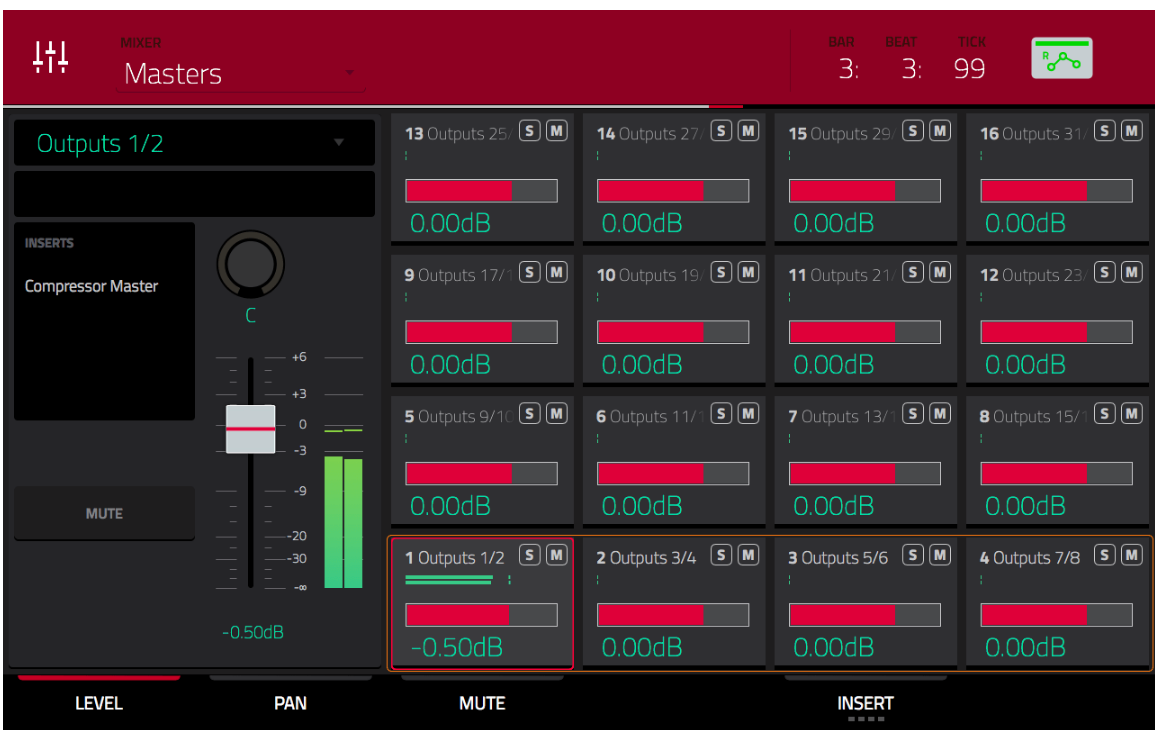
Here we have a dedicated master control for each set of MPC audio outputs. If you haven't adjusted any of the default output routings, then all audio generated within your MPC will leave via the main stereo outputs, known as **Outputs 1/2**.

Tap on **square 1** to select **Outputs 1/2**. Any mixing adjustments you make here in the MASTER channel will affect anything and everything intended for Outputs 1/2.

Hit **PLAY START** and watch the **LEVEL METER** on the left hand side. As you can see it is often heading into the 'red' zone which indicates that the signal is quite hot; it's possible that an overly hot signal is going to distort which is rarely a good thing when dealing with 'digital' audio.

We can just turn down the master LEVEL for outputs 1/2, however a better way is often to apply some subtle compression across the mix. Tap on the **INSERTS** box and add a **Compressor Master** effect with the following settings:

This applies a gentle, quick acting compression across the mix. Return to the MASTERS screen and observe the level meter.



You'll probably spot these two 'ghost' red lines at the top of the meter – these are there to show you the highest level the meter has reached in this session. Just single tap on these lines to reset them.

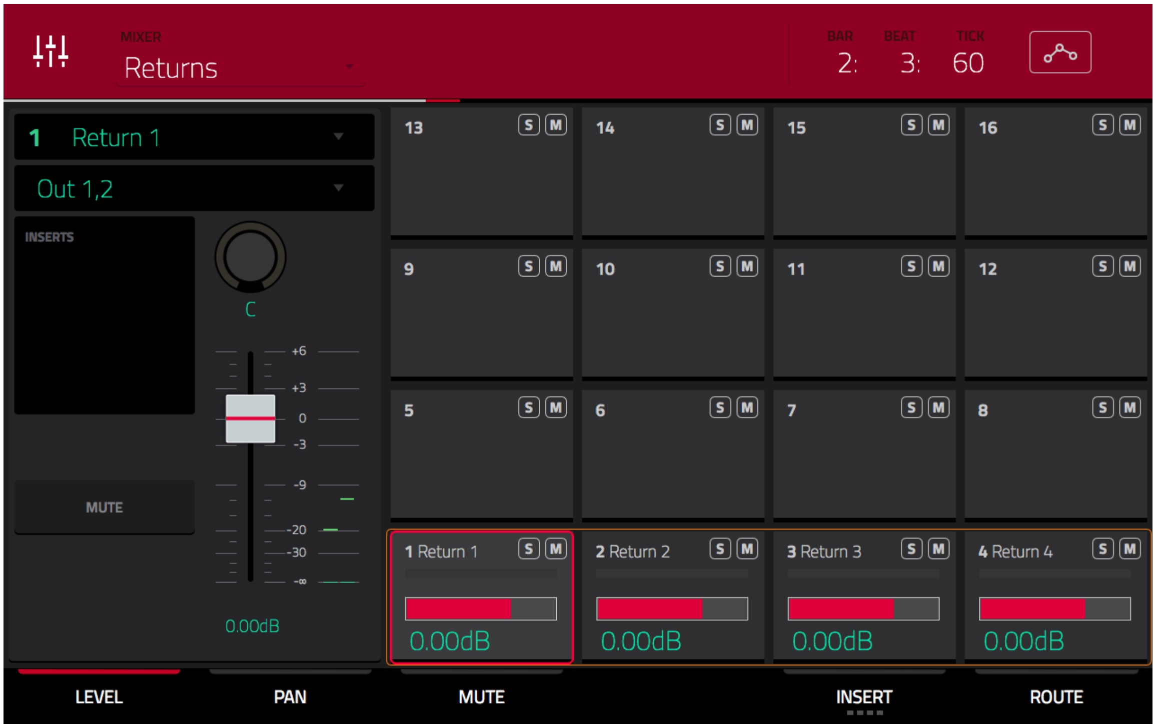
In addition to the compression, I also dropped the **LEVEL** to **-0.5dB** – at this point there was only a tiny bit of signal going into the red (the hi hat is the offender). Problems like this can be attended to as the mixing session (or sessions) progress. In this case I could just return to the PAD MIXER for the Shadow Kit V3 program and tweak the level for pad A03.

How to I Add EQ to a Mix?

The MPC mixer doesn't have dedicated EQ 'pots' like you get on a hardware mixer, so instead you have to use the internal EQ effect inserts; **PEQ 4-Band** and **PEQ 2-Band, 2 Shelf.** There are also a number of additional filter effects available (e.g. LP and HP Filters and Shelving Filters), so there's lots of options. To apply EQ across the entire mix, simply add the EQ effect to a spare MASTER INSERT.

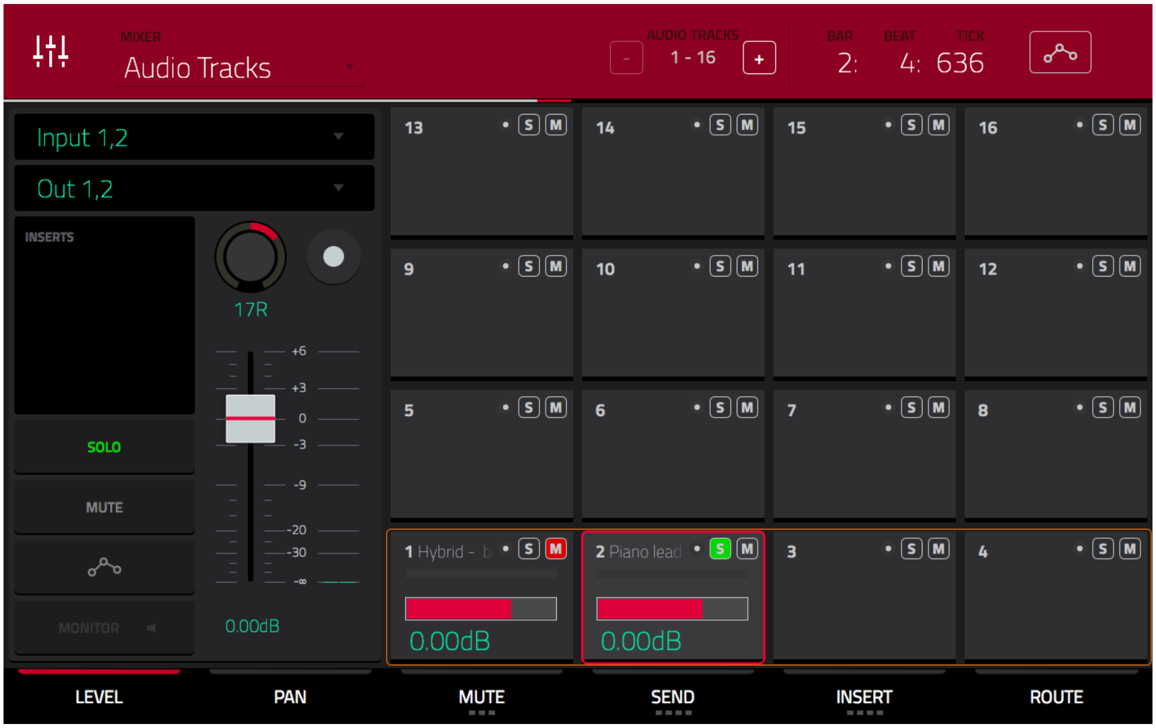
Using The RETURNS

When adding ambience effects like reverb to our mix, it is typical to add these via the 'send and return' system – this way we can add the same reverb preset instance to multiple sounds, which is more CPU efficient and also provides a more natural, consistent ambience across the mix . Go to **MIXER: Returns**:

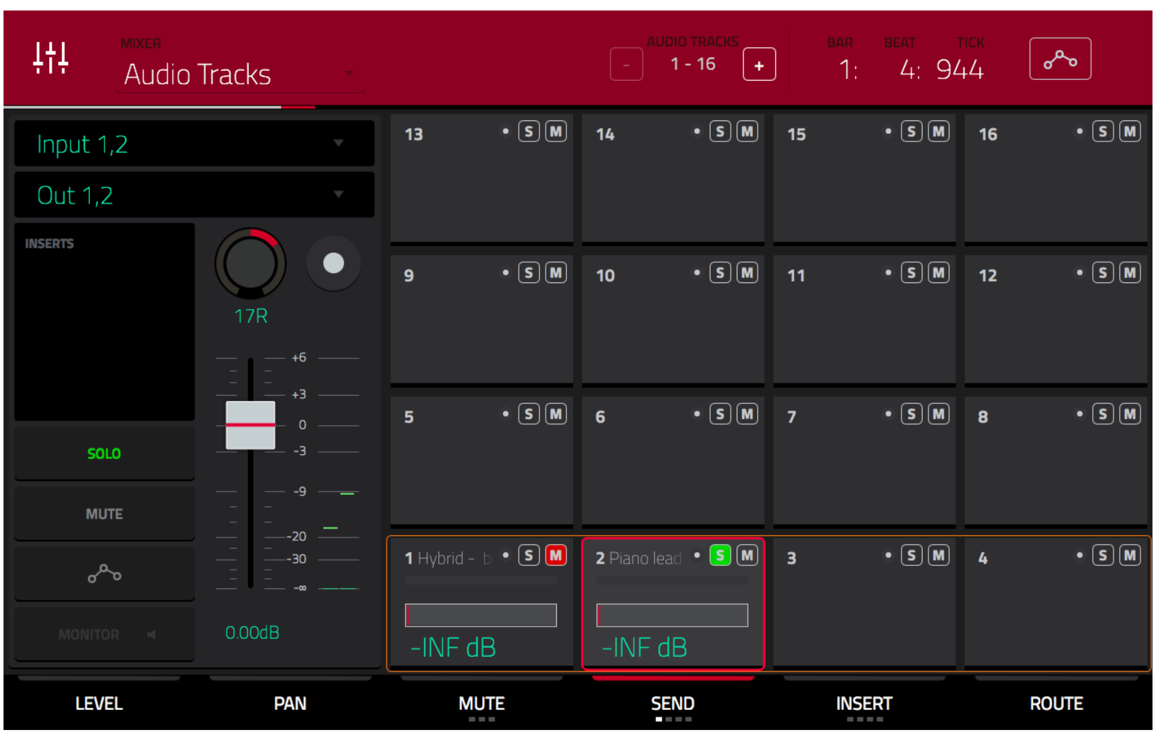


In the RETURN channel we can configure up to four different effects. The most common effects to use in a returns system are reverbs and delays, and perhaps chorus and phasers. Tap the Return 1 box to select it, tap the **INSERTS** box and add a **REVERB LARGE** effect; leave the reverb set to default parameters.

Let's add some reverb to our two audio tracks. In the **CHANNEL MIXER**, select **MIXER: Audio Tracks** and tap to select **audio track 2** (**Piano Lead**). Press the **SOLO** button so we can only hear this track. Press **PLAY START** to hear the audio track play back.



Now tap the **SEND** button at the bottom of the screen.

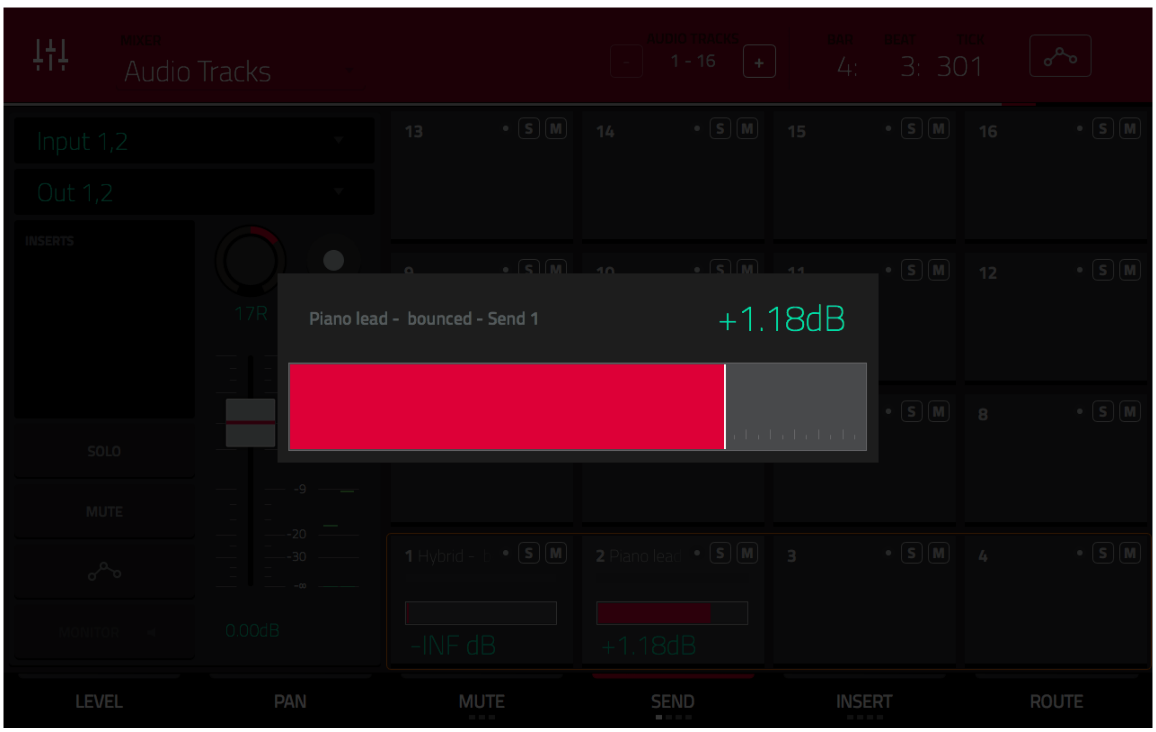


In the SEND screen we control how much effected 'return' signal is added to the currently selected track. You can see the SEND button has four dots under it –

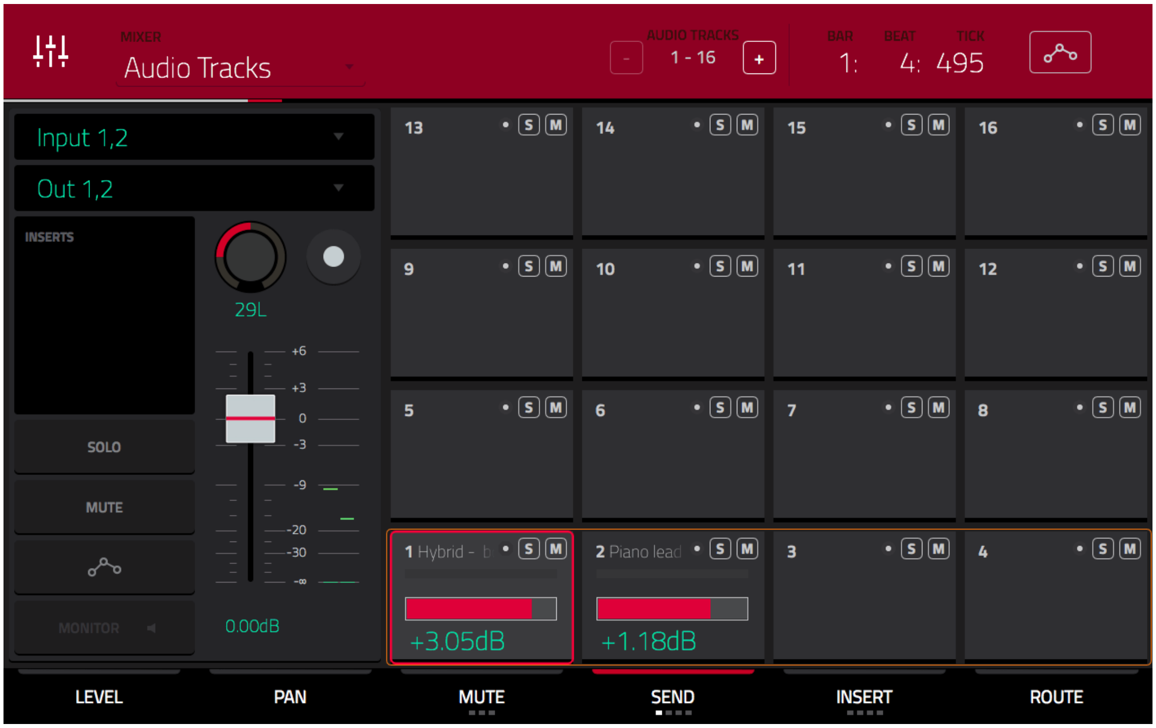
this reflects the fact that we have four returns, so as we added our reverb to **RETURN 1**, make sure the first dot under the SEND button is active (**SEND 1**).

To add some reverb to our audio track, so increasing the **SEND** value from **–INF dB**. As you increase the value you should eventually start hearing the reverb being added to the track. The more reverb you add, the more 'distant' the piano becomes.

Hit the **SOLO** button again so you can hear the piano play with the rest of the beat and continue adjusting the **SEND** value until you are happy. I set mine to **+1.18**. Remember you can double tap the small slider to bring up the full screen slider:



Perform the same procedure for audio track 1. Remember sounds with less reverb sound closer in the mix, sounds bathed in reverb sound very distant. I set the **SEND** for **audio track** 1 to **+3.05**.



You can check out my draft mix by loading up the **Mixing-Draft.xpj** project from the **chapter 013** folder.

Dealing With The Submix Omission

As we know, there's no submixing in standalone mode and this is a pain. If you do set up submixing in controller mode, those submixes are ignored and throw up an error when loading that same project in standalone mode.

So let's say you wanted to send your drums to a separate submix to apply a compressor and some global level control to the kit – how would you do this without any submixes?

The best method is probably to use the **program channel** where you can apply a compressor across the whole program and also tweak the global level of the all pads in the program. The problem we have here is that our 'Shadow Kit' program also contains a bass sample on pad A05 so any effects and level changes we apply to our drum kit will also be applied to our bass pad.

The solution is to take the bass note out of our Shadow Kit program and place it in a separate 'bass' program. This means reassigning the bass track to play the different bass program, but that's not a big deal when you are only having to change a single sequence.

The key is to plan ahead with this issue. In the final two chapters I'll be working with a song that from the outset used a dedicated program purely for anything I'd want to run through a submix (in this case, the drum kit). However it's definitely not ideal having no submixes available, so let's hope Akai bring submixes back into standalone mode sooner rather than later!

Check out **sequence 2** of the **Mixing-Draft.xpj** project ('**No Submix'**). Here I've split the Shadow Kit program into two new programs, '**Just Bass'** and '**Just Drums**' and reassigned the bass track 2 to the 'Just Bass' program.

To create the two new programs I just hit the pencil icon on the **PROGRAM** row in **MAIN** and selected **DUPLICATE** – this duplicated the Shadow Kit program – I then repeated this again. I then renamed these two programs and edited the pad assignments in PROGRAM EDIT accordingly.

In this example I've now used the **CHANNEL MIXER > PROGRAMS** to insert a compressor and an MPC3000 vintage effect across the 'Just Drums' drum kit.

014 Working with Loops

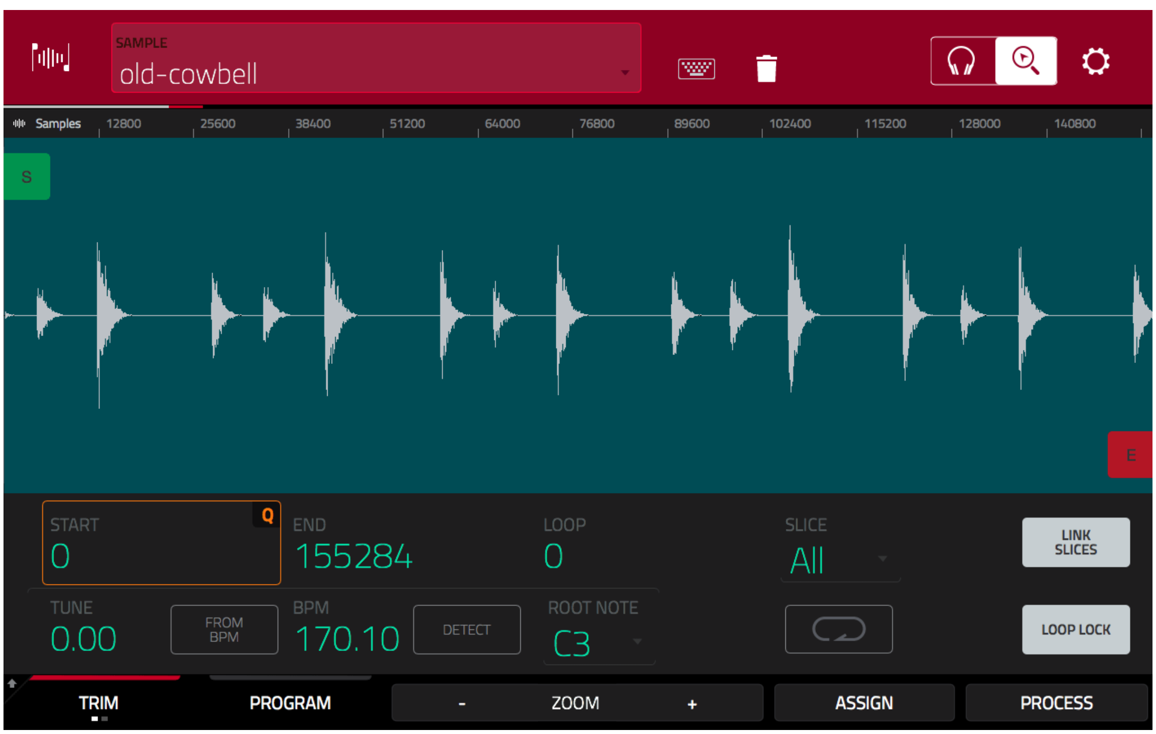
A loop is a very handy type of sample that is configured to repeatedly play between two selected points. Loops are often short phrases such as drum breaks or musical riffs but can also be used to extend the length of individual musical note samples.

How Do Loops Work?

A looped sample is one that can just *seamlessly* repeat over a defined portion of that sample. The simplest type of looped sample is one that loops from its very beginning to its very end and can be repeated infinitely to produce a continuous performance.

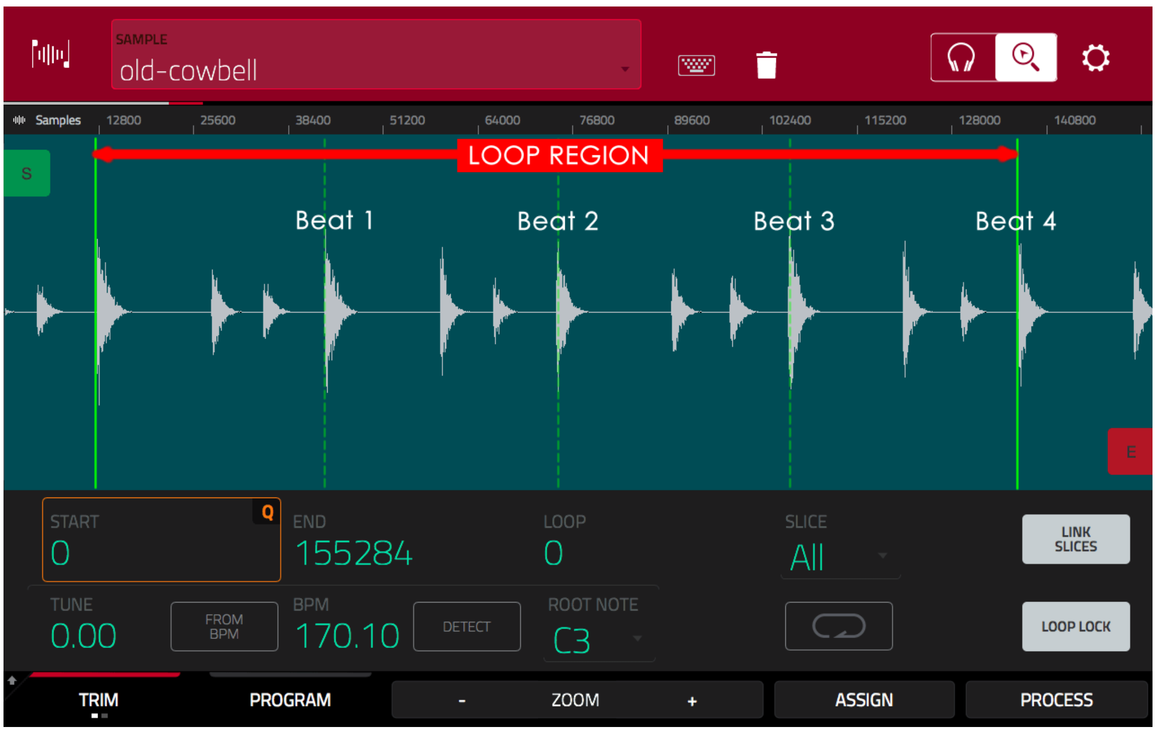
If you've ever wallpapered a room with a patterned wallpaper you'll know that each strip of wallpaper must be placed so the pattern matches the previous strip, forming a continuous seamless pattern across the wall. Looping audio samples is no different. In this tutorial we're going to loop a percussion sample.

Load up the sample **old-cowbell**; this is a short recording of a live cowbell performance. Let's have a look at it in **SAMPLE EDIT**:



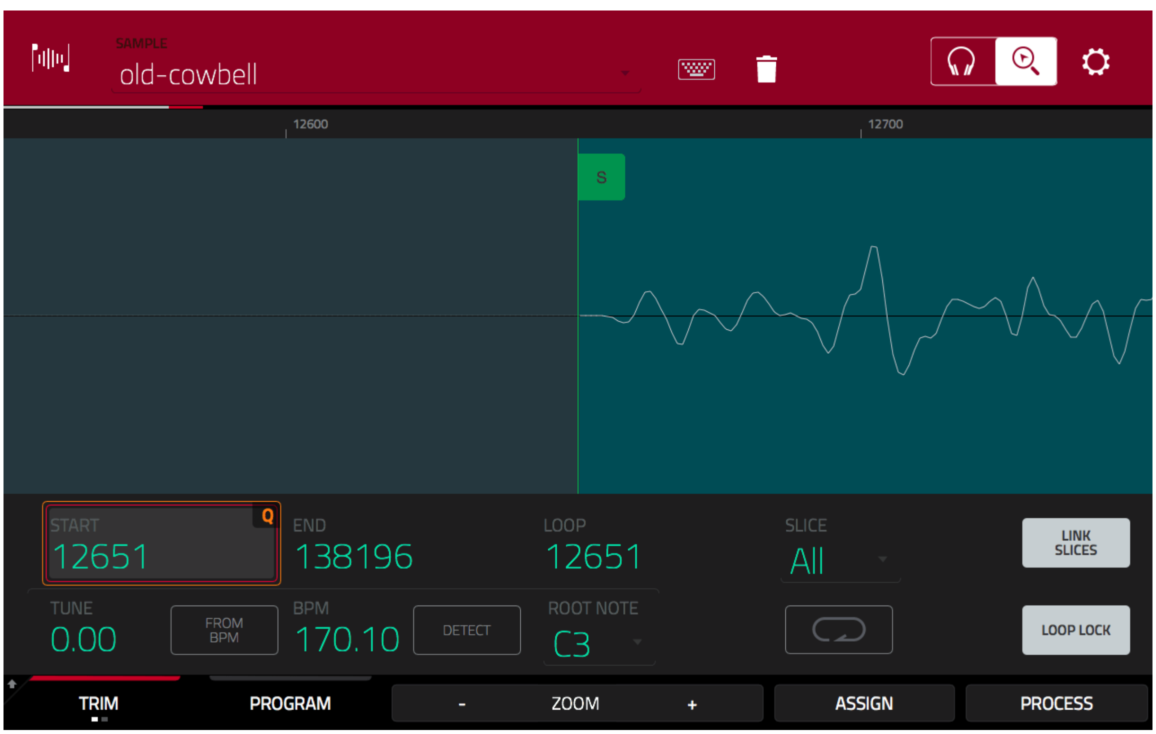
Hit **pad 1** to preview this sample from start to finish; hopefully you can hear that at the moment this isn't what we would normally call a perfectly *looped* sample, although I suspect you can already recognise a portion of the sample that is crying out to be looped.

When looping music we tend to loop them in 'whole bar' portions, often 1 or 2 bars to keep them manageable. In 4/4 time signatures, a bar consists of 4 beats (count: 1, 2, 3 and 4). To my ears, the obvious looped portion for this sample needs to be in the following region, which is 4 beats, hence 1 bar (again, count the 4 beats as the sample plays through:

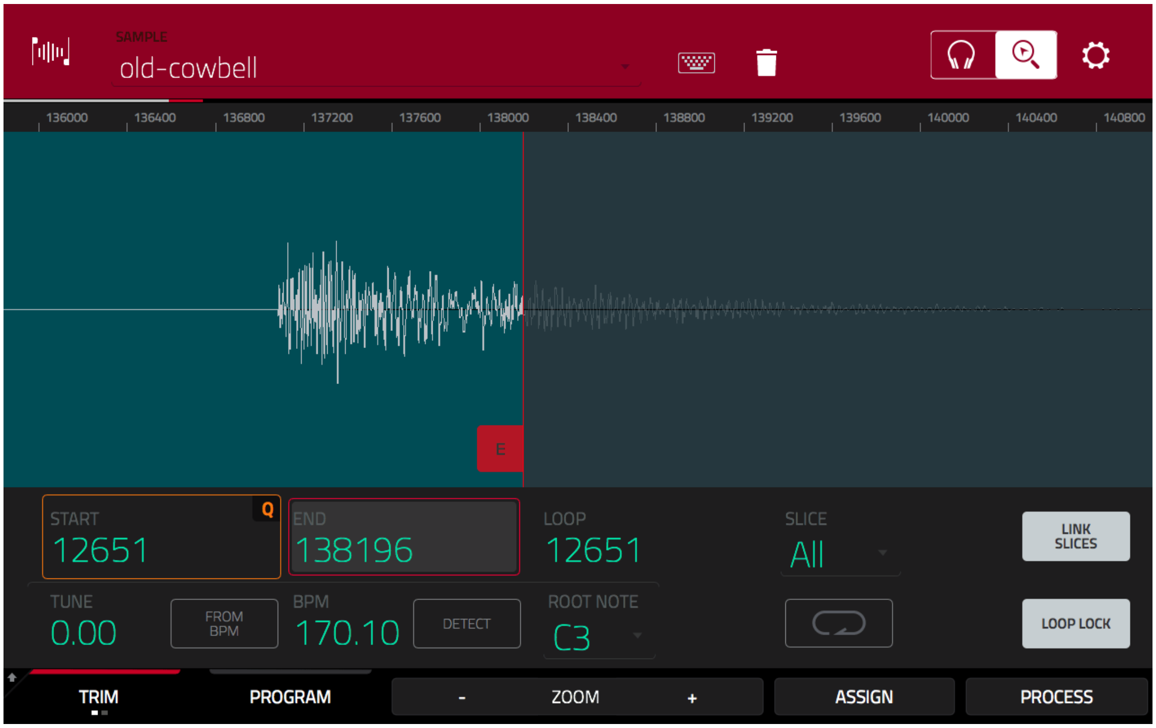


Firstly let's set the rough start and end points for our loop. Hold down **SHIFT** and set **SNAP 0** 'on'. Using the techniques discussed in the sampling chapter, set your start and ends points similar to the following with the **START** at around **11656** and the **END** at around **138196**:

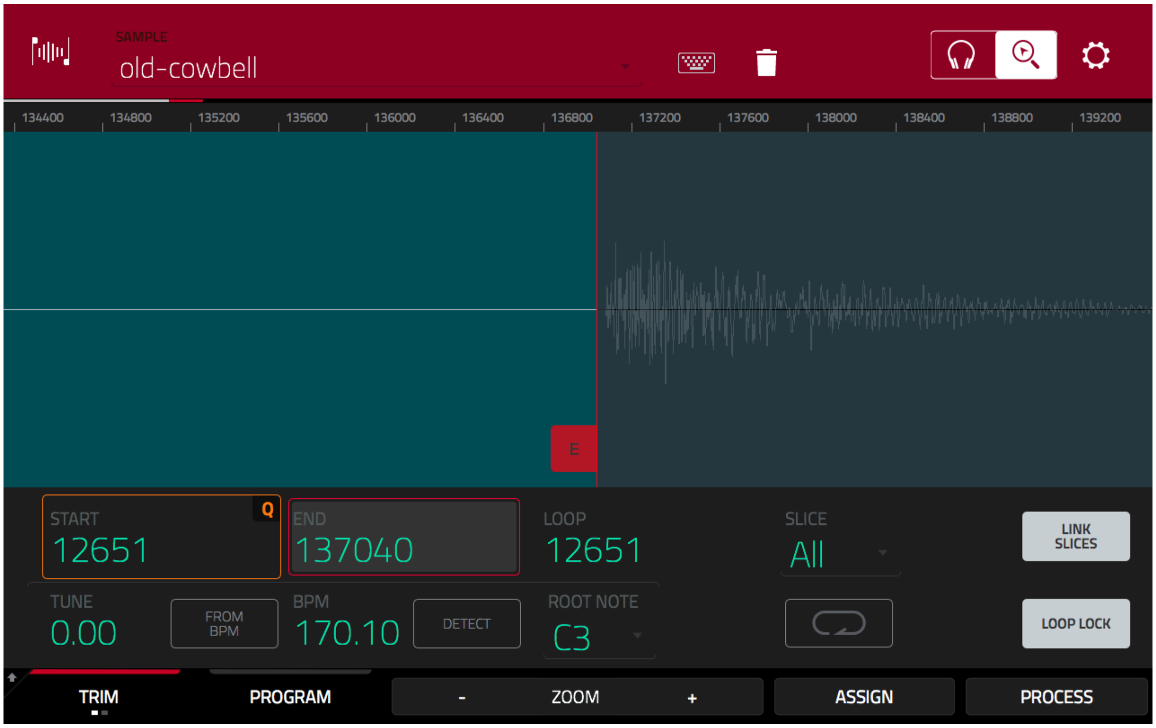
Hit pad **A10** to preview. This is very rough – the beginning is a little late and the end 'clips' a cowbell hit. First, let's get the start nice and 'tight', so pinch and zoom the waveform and drag the waveform and increase the **start** point to **12651** – remember, **'SNAP 0'** will ensure we have the edit points perfectly on zero crossing points:



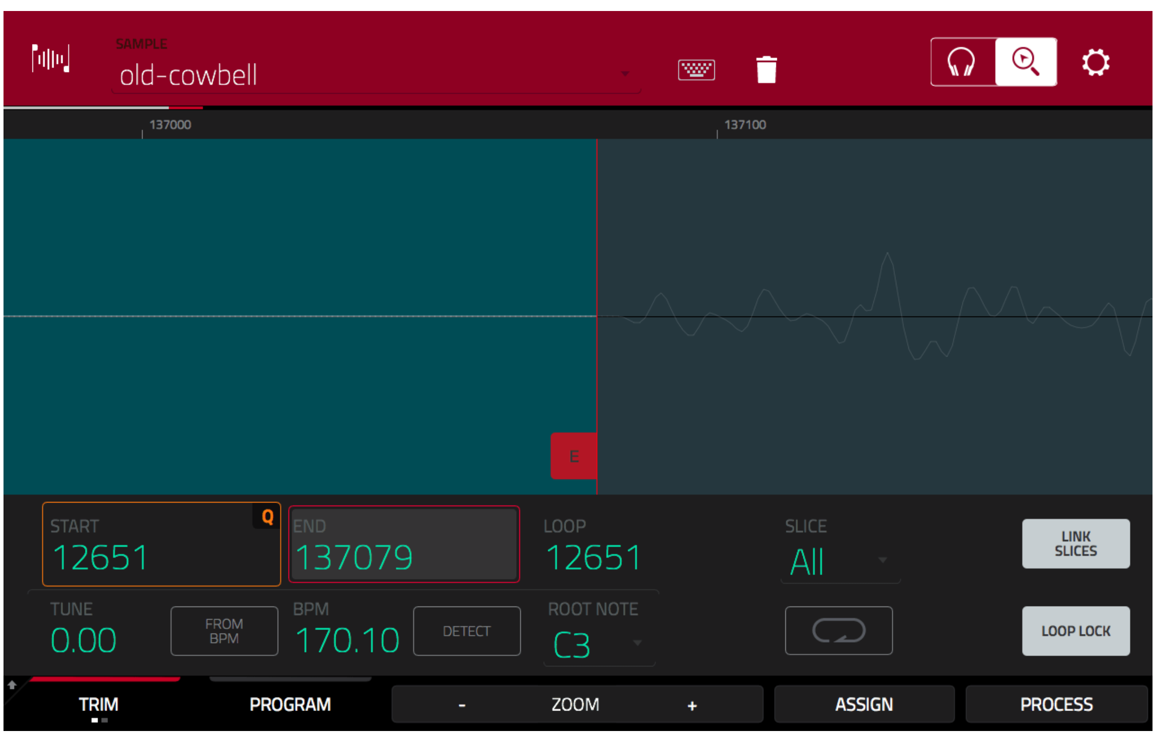
Preview this edit by briefly pressing down and holding the ‘note on, play from start’ preview, **pad 5**. Sounds good to me, so let’s move on to the **end point** of our loop. Initially, set a magnification level that allows you to easily see that cowbell hit that we are currently clipping:



Take the end point back until you are positioned just before that large waveform peak (approximately **137040**):



Now pinch and zoom in and make the final edit of the end point. I found an end point of **137079** appears to be spot on:

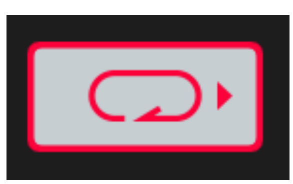


Notice how, visually, the waveform cycle at the end point 'matches' the waveform at the start point – you can see this by tapping back and forth between the

START and END point parameters on screen. Here's a single graphic that shows how the two edit points match up:

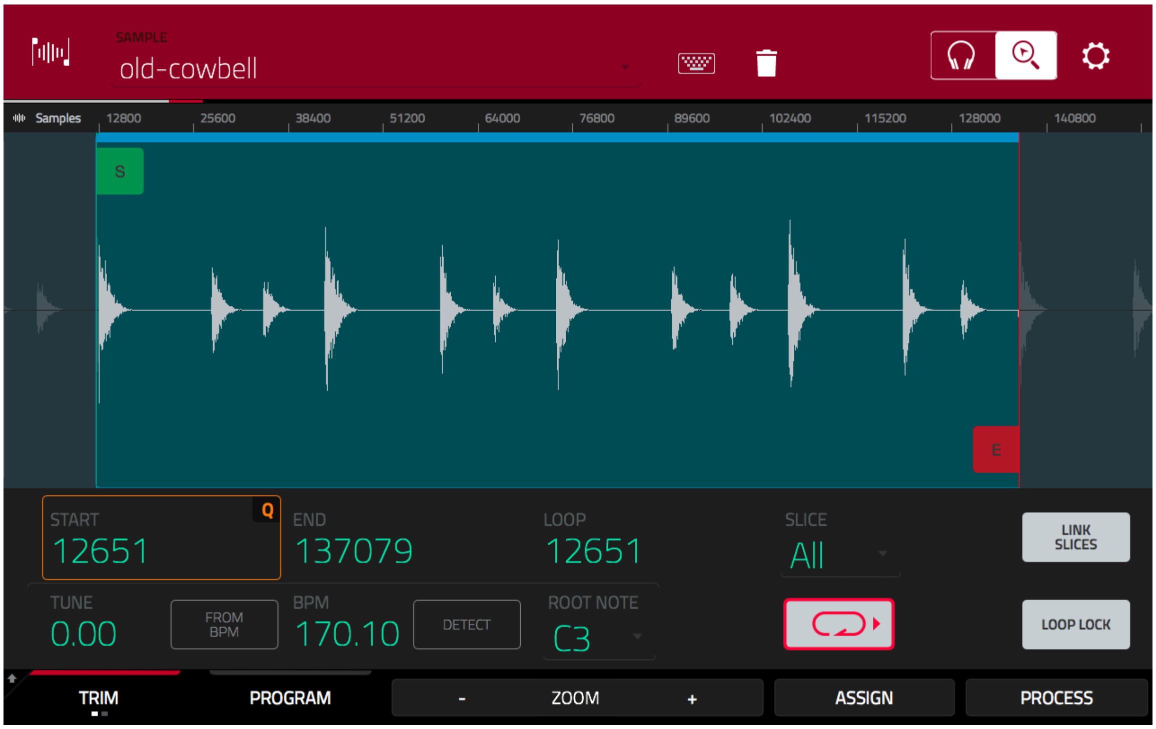
To preview what our 'loop' now sounds like, press and hold pad **13** (**Play Loop**) and your break will continue to play until you release the pad. Alternatively, just hit **pad 16** (**Continuous Loop**) and you'll hear your break 'loop' continuously until you hit pad 16 once more. As a 'loop' it sounds good to my ears, the timing sounds perfect and there's no audible clicks.

Now, currently the MPC is only looping this sample when we specifically use the **Play Loop** preview pad (**pad 13**). This means that when we use the sample in one of our beats, it will not actually *play* looped. Hence we first have to 'officially' tell the MPC that we want to use this sample as a 'loop'. To do this, we need to set the **LOOP** parameter from OFF to **'Forward'**. Press the '**LOOP'** parameter once so it turns white:



The little arrow pointed to the right indicates that we've set this sample as a 'forward loop'. A forward loop will loop our sample continuously in a normal, forward direction (i.e. from start to end, then immediately back to the start again).

Upon setting the sample as a loop, you'll see a blue bar appear at the top of your waveform, indicating the sample is looped:

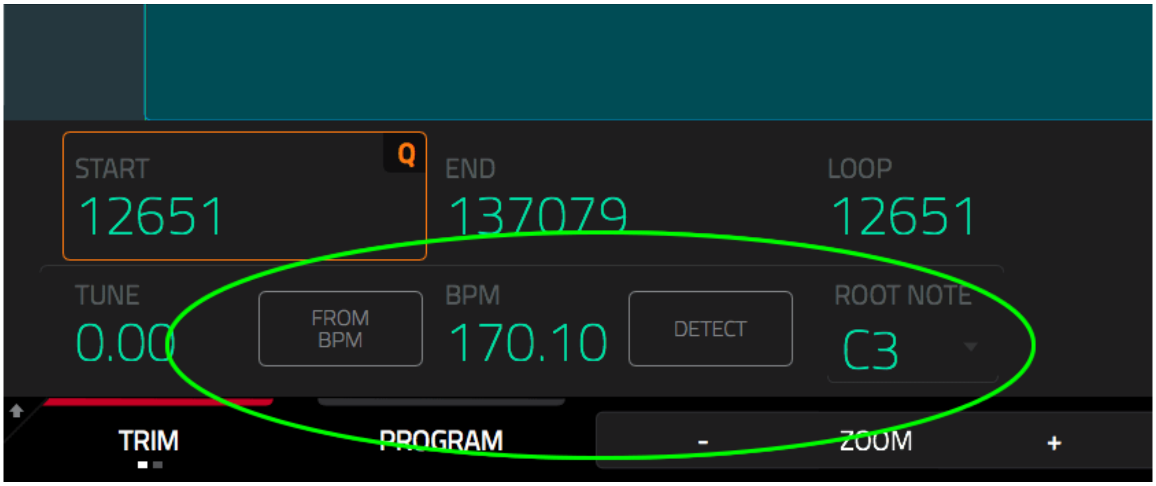


Now press and hold **pad 11** (**Play Sample, Note On**) – the sample continuously loops in normal play, thus indicating it is now assigned as a 'loop' sample. You can now, if you wish, press **PROCESS** and '**DISCARD'** any unwanted audio either side of the start and end point.

If you wish, load up my version of this looped cowbell sample – **old-cowbell-looped.wav**.

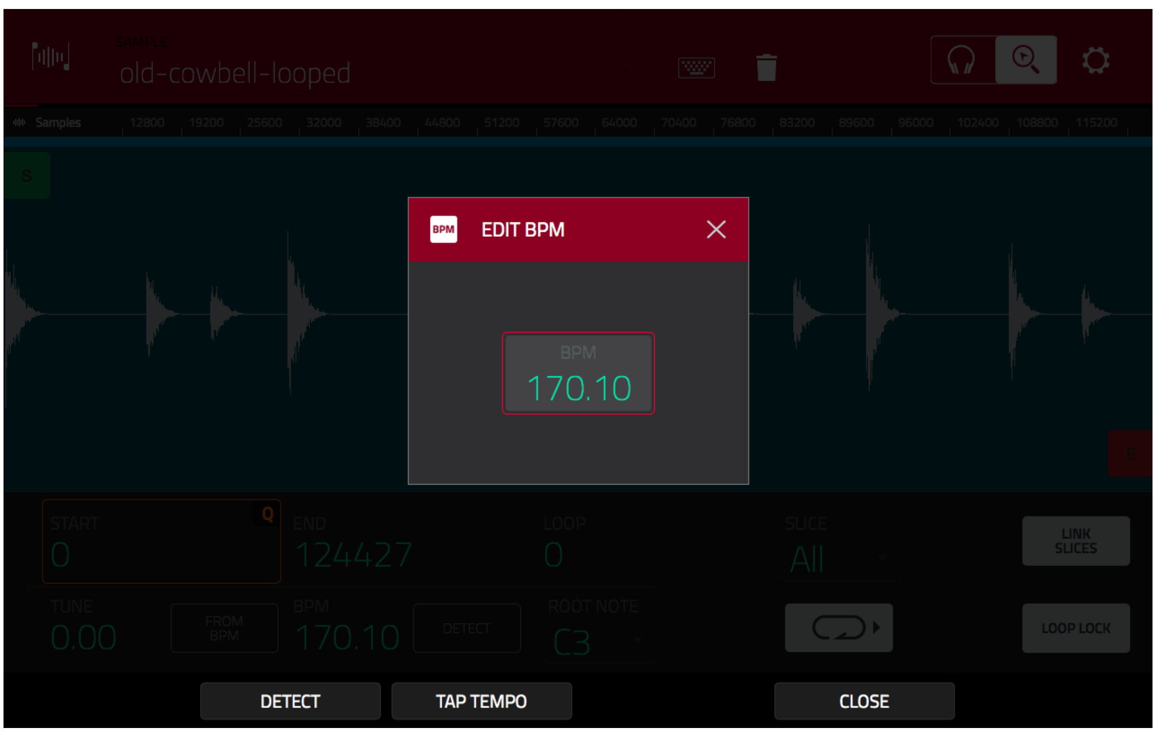
Setting Loop Tempo

One of the most important settings we need for a loop is to calculate its **tempo**. The MPC actually already attempts to do this for you in the SAMPLE EDIT screen:

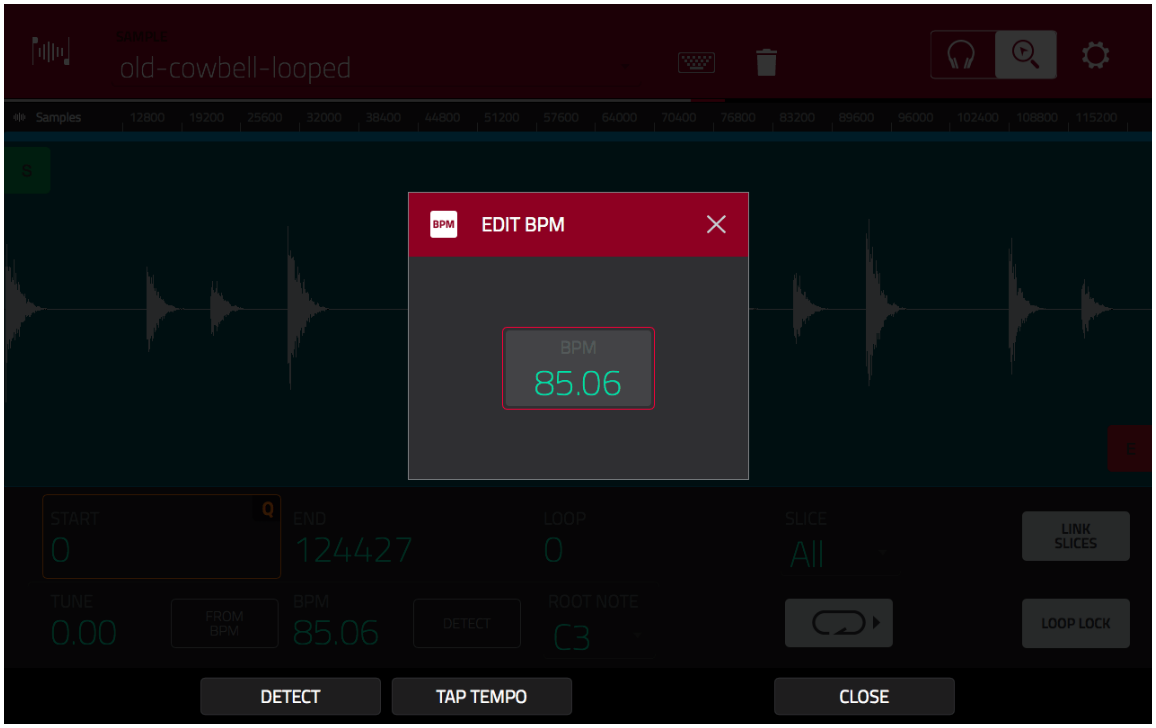


However, a tempo of 170.10 BPM seems very fast for this particular loop. This is because the MPC tends to initially assume all loops are 8 beats (2 bars) long and just performs some simple maths from the sample length to calculate the BPM.

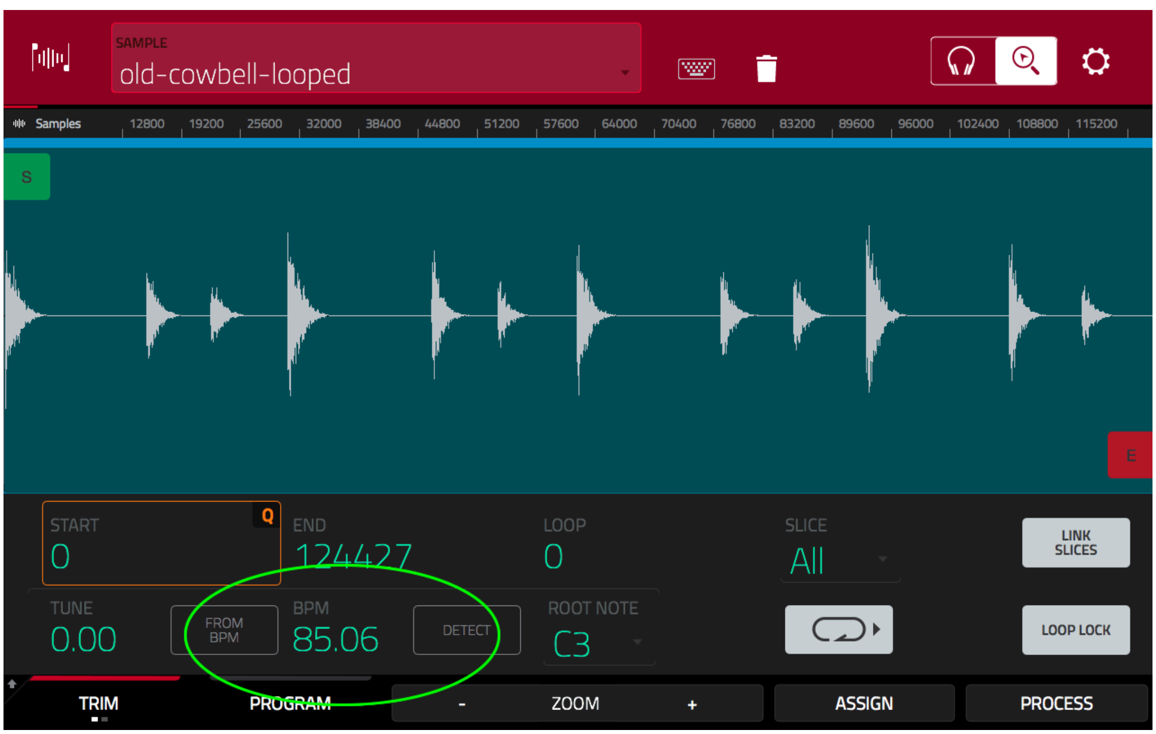
A more reliable method can be obtained from using the **DETECT** option. Hit **DETECT** to bring up the **EDIT BPM** screen:



At the bottom of the page is the **DETECT** button – hit this and the MPC will analyse the transients in your loop and will (hopefully) come up with a more accurate calculation of the loop tempo:



That seems more correct, and it's pretty much half the tempo of the previous calculation which makes sense (as this is a 4 beat loop, not 8). Return to the main **SAMPLE EDIT** to see the updated BPM reflected on screen:



The MPC is pretty good at detecting the correct BPM, but if you know it's made a mistake you can override this BPM setting if you wish – just enter a different BPM by double tapping the BPM field. Or hit **DETECT** and continually tap the on-screen **TAP TEMPO** button at the tempo you wish to set it to (good luck with that one!).

Using Loops in a Sequence

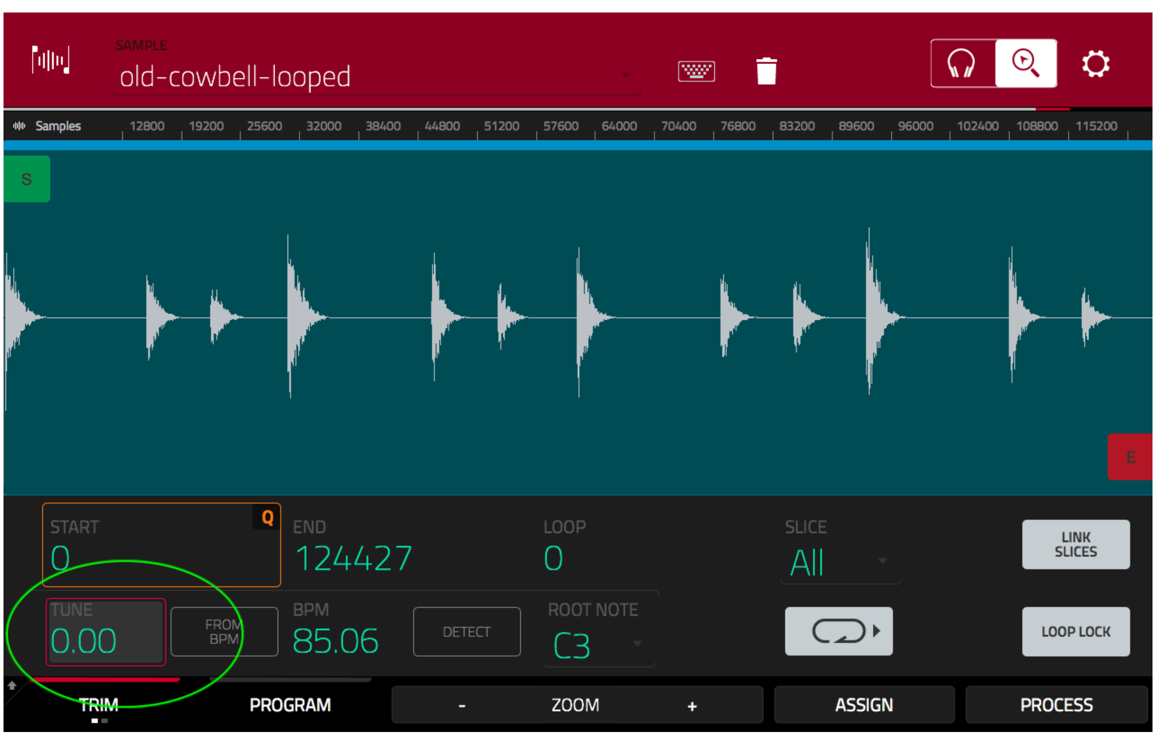
One of the defining features of a loop is that, unlike 'single hit' samples it has a distinct 'tempo' and therefore any sequence you wish to use it in must have a tempo to match (or vice versa!).

We know that our cowbell loop is 85.06 BPM. Now, the 'Demo Beat' sequences we've been building in the examples so far are set at 90.0 BPM, so in order to use this cowbell loop in any of those sequences, something will have to change. That could be the tempo of our loop, it could be the tempo of the sequence itself, or we could even change the tempo of both the loop and the sequence.

I'll look at changing sequence tempo in the Warping chapter; at this stage let's concentrate on some methods we can use to change the tempo of the loop itself.

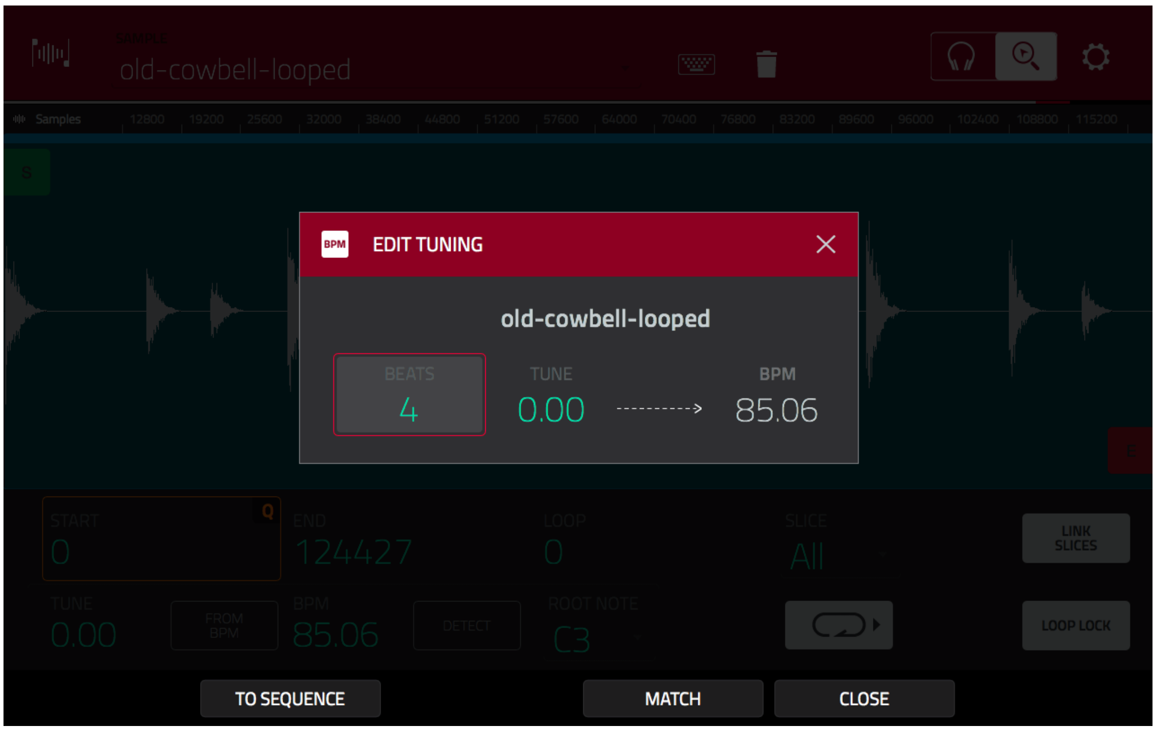
Re-Tuning The Loop

In **SAMPLE EDIT** make sure you've selected the **old-cowbell-looped** sample. At the bottom left of the screen is the **TUNE** parameter.



Changing the *tuning* of a sample will alter its musical pitch. However, in the case of a percussive loop we are more concerned with how changing the pitch will change its tempo. If we lower the TUNE value (i.e. lower the pitch) we will be slowing down our sample. If we *raise* the TUNE parameter, we will be *speeding up* our sample. Therefore, we can use TUNE to change the tempo of our loop to match the tempo of our sequence.

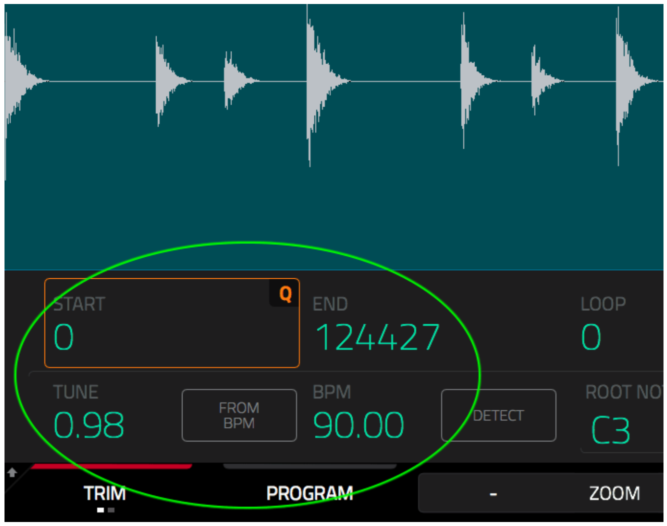
Hit the **FROM BPM** button:



In this screen you can adjust the tuning of the sample and watch how it changes the BPM. Make sure you've selected **'4 BEATS'** so the current tempo shows correctly (85.06).

At the bottom of the screen we have two options; **TO SEQUENCE** and **MATCH.** MATCH will automatically tune our sample so its BPM matches that of our current sequence. **TO SEQUENCE** this will find the 'best fit' compromise in cases where the MPC is unable to perfectly match the sample tempo to current sequence tempo. In this case it will get the sample tempo as close to 90.0 as possible and then adjust the sequence tempo slightly to match them together.

We currently want to avoid changing sequence tempo and instead concentrate on only changing the loop itself, so hit **MATCH**. You'll see the MPC has tuned the sample to **90.0 BPM**

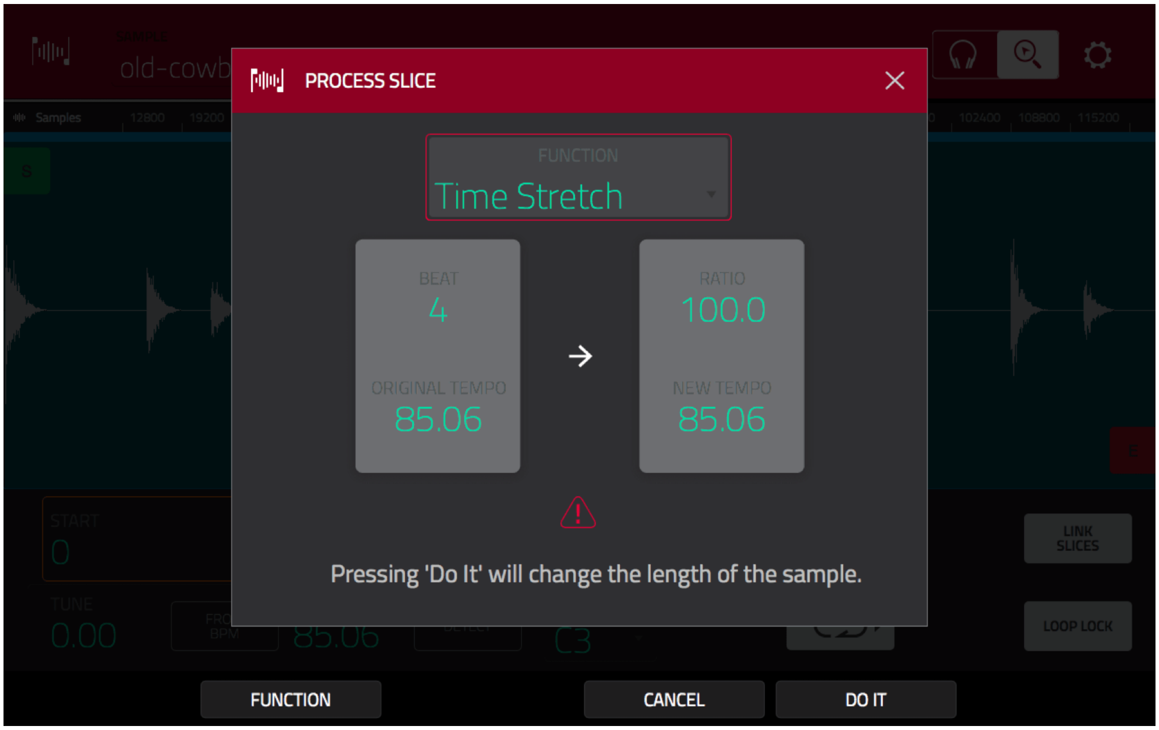


Re-name this sample **old-cowbell-tuned**.

Time Stretching a Loop

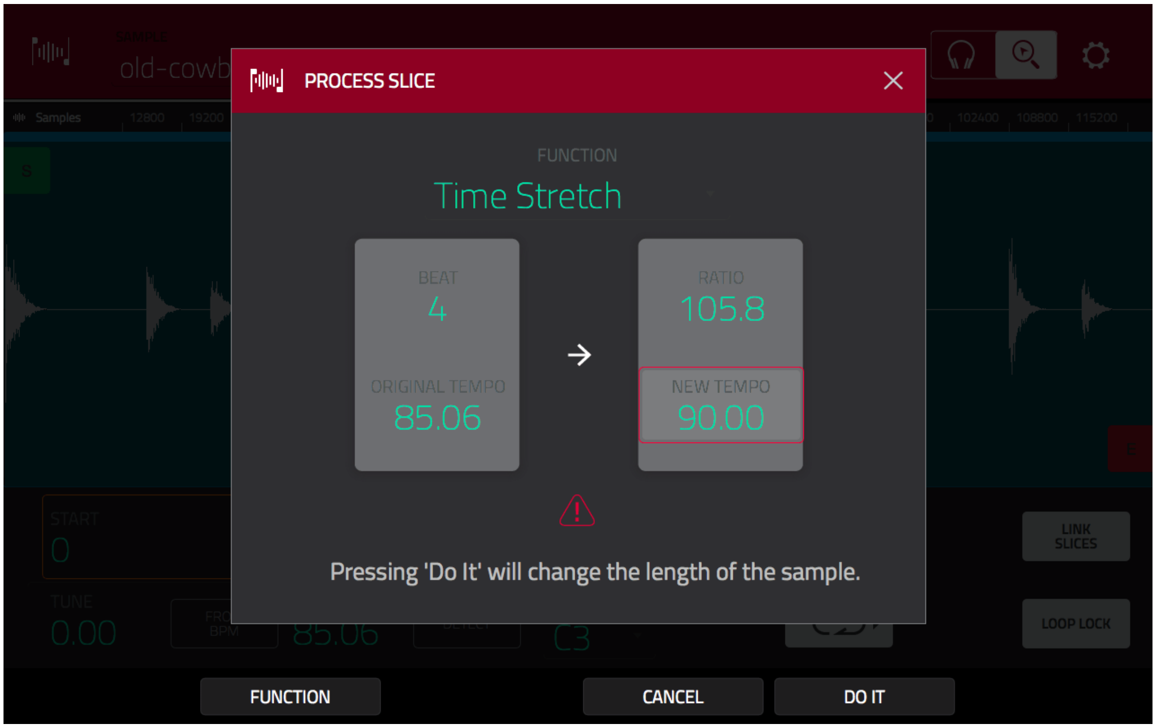
As you can probably hear, tuning this loop has changed its musical pitch – it clearly sounds higher pitched. This is sometimes not a problem with a drum or percussion loop, but it's usually undesirable to change the musical pitch when dealing with musical loops (e.g. pianos, bass, vocals, guitar etc).

To avoid tuning our sample, we can instead look at time stretching. From the BROWSER, load up the original **old-cowbell-looped** sample from the **chapter 014 folder**. In **SAMPLE EDIT**, go to **PROCESS > Time Stretch**:



Timestretching is an algorithmic process that attempts to change the length of a sample without changing its pitch. It has its limitations, especially when trying to slow down samples by a large amount, but it usually works very well for small changes (especially speed increases).

Single tap the **NEW TEMPO** field, hold down **SHIFT** and turn your data wheel to set a target tempo of **90.00**.



Hit **DO IT** to return to the main **SAMPLE EDIT** screen.

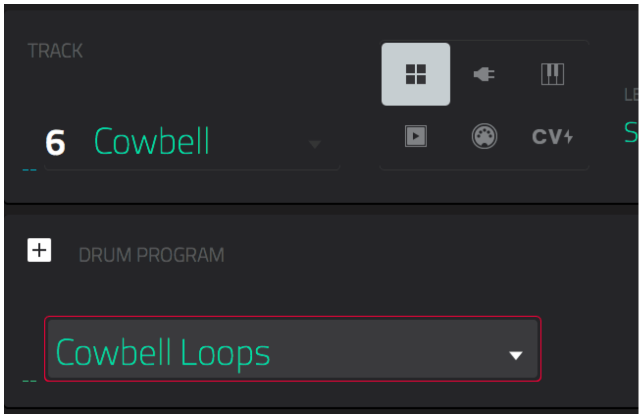
Now there's a little bug here as the MPC still initially shows the tempo as 85.06, but if you hit **DETECT > DETECT** you'll see the MPC finally recognises that it's changed the tempo to 90.00.

Preview the timestretched loop – rename it **old-cowbell-timestretched**.

Using Loops in a DRUM program

Let's add our cowbell loop to our ongoing 'Demo Beat' sequence. From the **BROWSER** load up the **Loops.xpj** project – this load up our current '**Demo Beat**' sequence to **sequence 1** – the only difference is I have muted the piano chord track to free up some room for our cowbell loop.

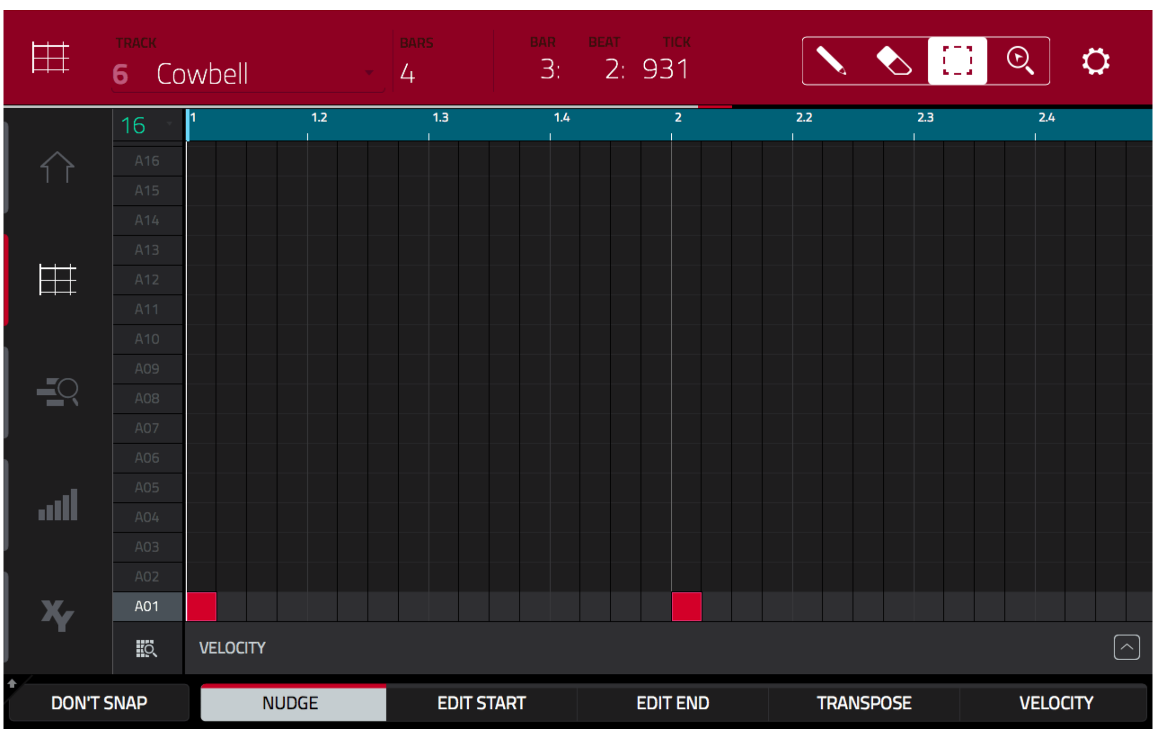
In **MAIN**, make sure **MIDI** tracks are selected and hit the **TRACK+** button until you find the next unused MIDI track. Select a **DRUM** type and create a new DRUM program called **Cowbell Loops**. Rename this track to '**Cow Bell**'.



Hit pad **A01** and go to **PROGRAM EDIT > SAMPLES** and assign old-cowbell-tuned to layer 1. Now when triggering loops from a DRUM program we have two distinct options. The loop can be triggered as a ONE SHOT or NOTE ON.

If we set the pad to the default ONE SHOT, the loop will play the entire bar from start to end, but then it will stop. This means that in order to cover four bars, the loop will need to be triggered four times, each one at the very beginning of each bar of our sequence.

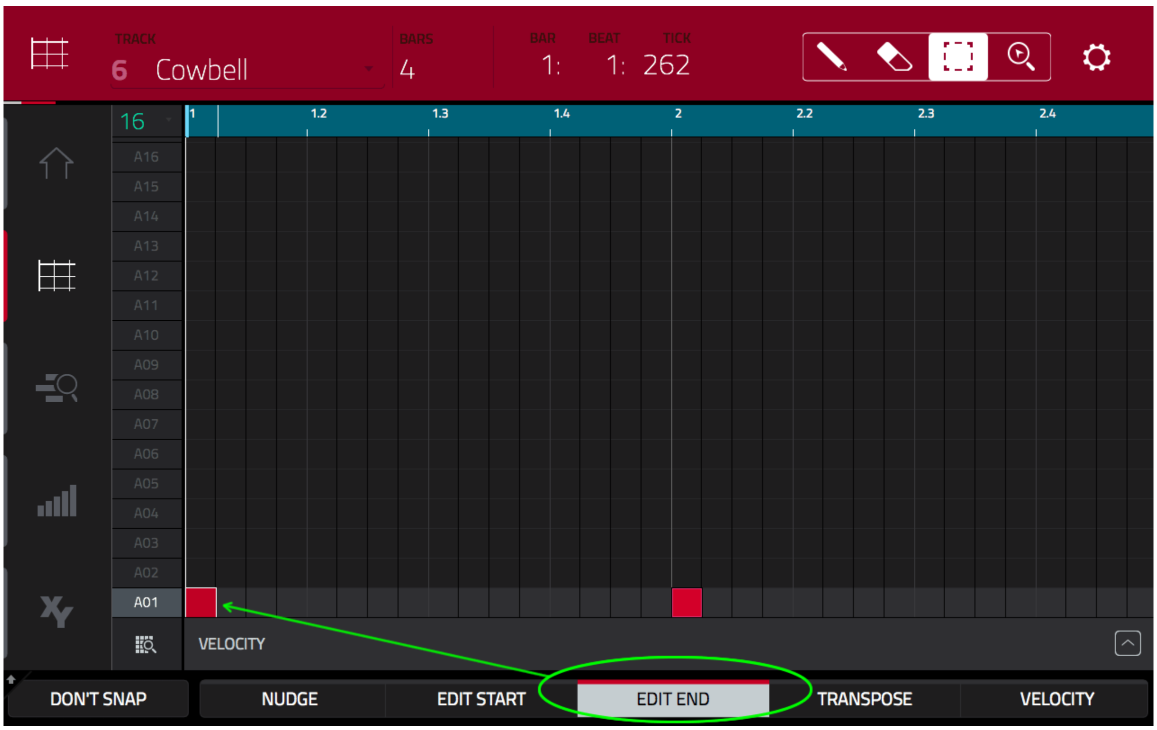
Now using the **GRID**, insert an **A01** event at the beginning of each bar in our sequence:



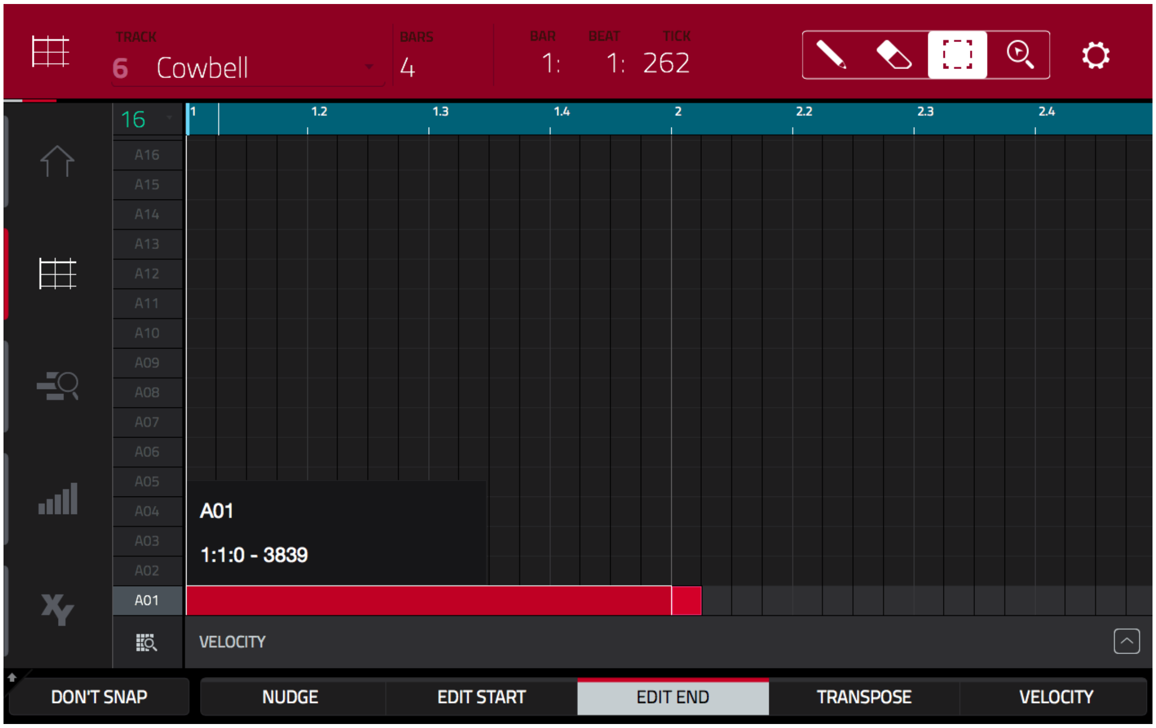
Press **PLAY START** to hear the tuned cowbell loop in action.

Alternatively, we can set up the cowbell loop to play as a NOTE ON sample. Go to **PROGRAM EDIT > LFO MODULATION** and set **SAMPLE PLAY** for **A01** to **NOTE ON**. Hit pad **A01** and, as expected, the cowbell loop only plays for as long as you hold down the pad. If you press **PLAY START** on your sequence now, you'll only hear a single cow bell hit for each cowbell loop instance – this is because the duration of the events in our sequence are not very long (they didn't need to be for a ONE SHOT).

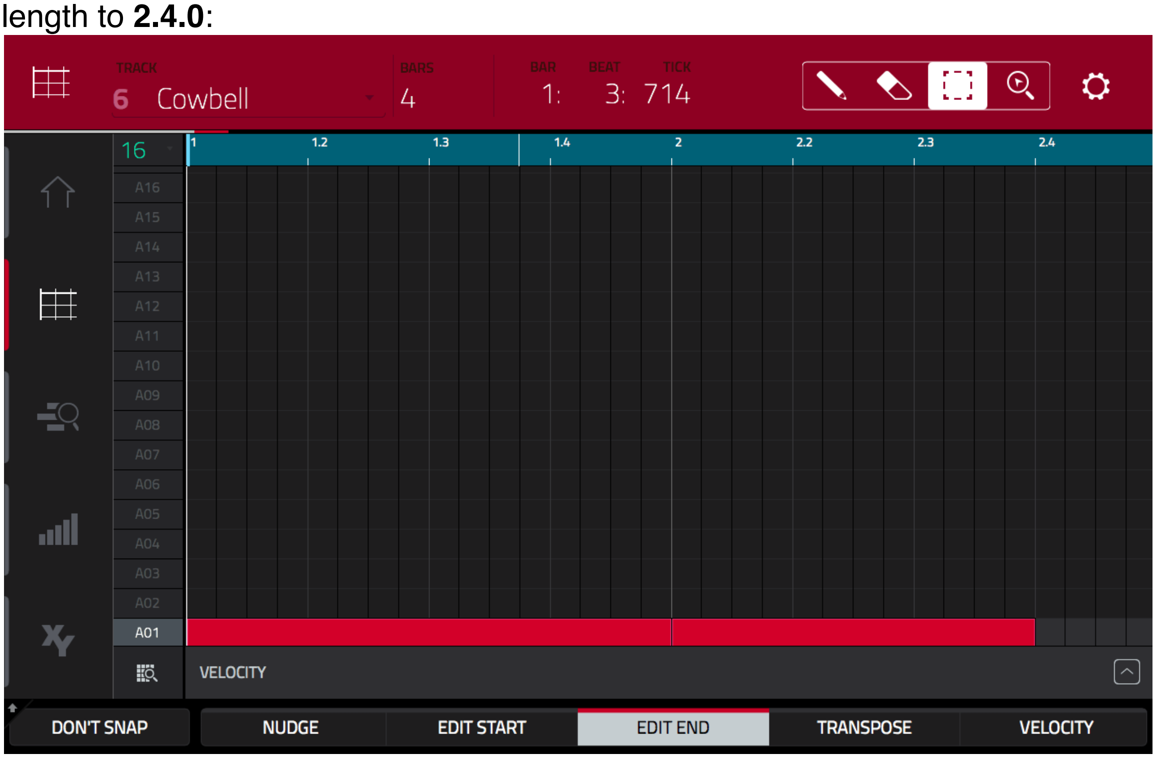
To insert our loop as a NOTE ON event, we need to ensure the event has a suitable duration. In **GRID**, choose the **SELECT tool** and select the first cowbell loop event at **1.0.0**. Now press the **EDIT END** button at the bottom of the screen:



With **EDIT END** selected, turn your data wheel clockwise to extend the length of the selected event – take the event all the way to the end of the first bar:



Do the same for the second event at **2.1.0** but this time only extend the event length to **2.4.0**:



Repeat this for the remaining two events, with the **bar 3 event** extended all the way to the end of the bar and the **final event** extended to **4: 3: 0**. Press **PLAY**

START – as you can hear, NOTE ON mode allows us to dynamically vary the length of the loop so we can cut it short whenever we wish.

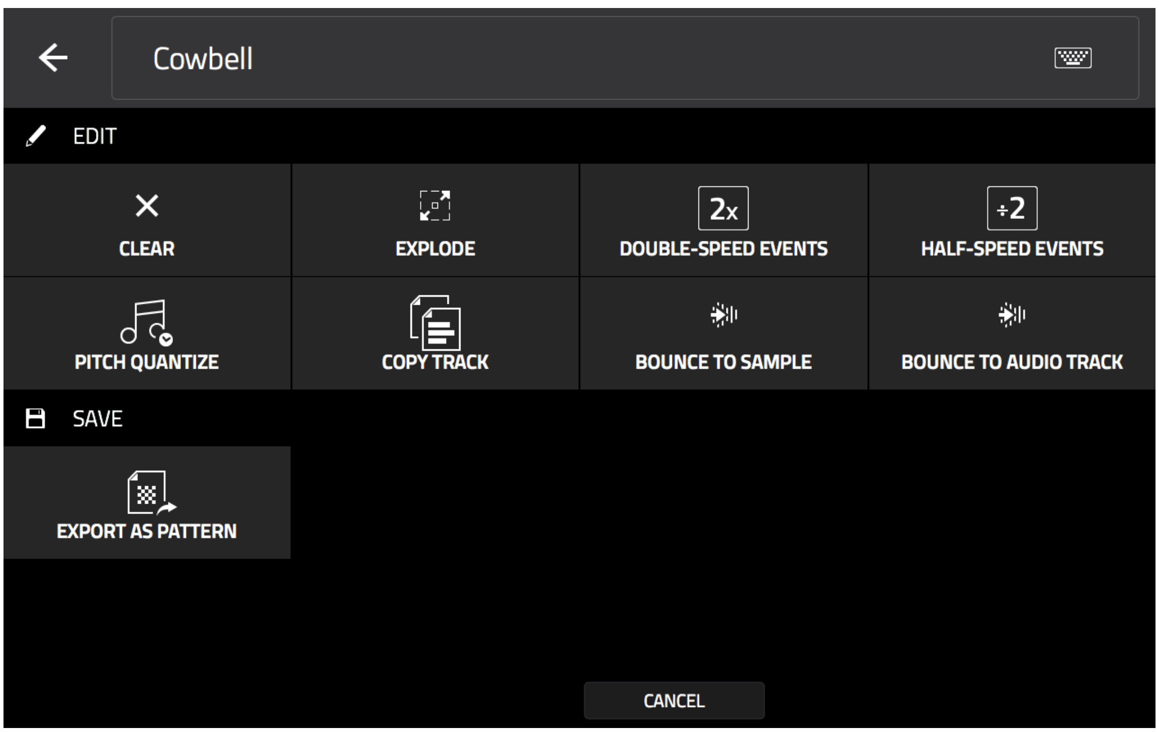
Another advantage of using NOTE ON loops is that the moment you press STOP in your sequencer, the loop also stops. ONE SHOT events will continue to play to their end (unless you hit STOP twice).

You can hear my version in **sequence 2** of the **Loops.xpj** project.

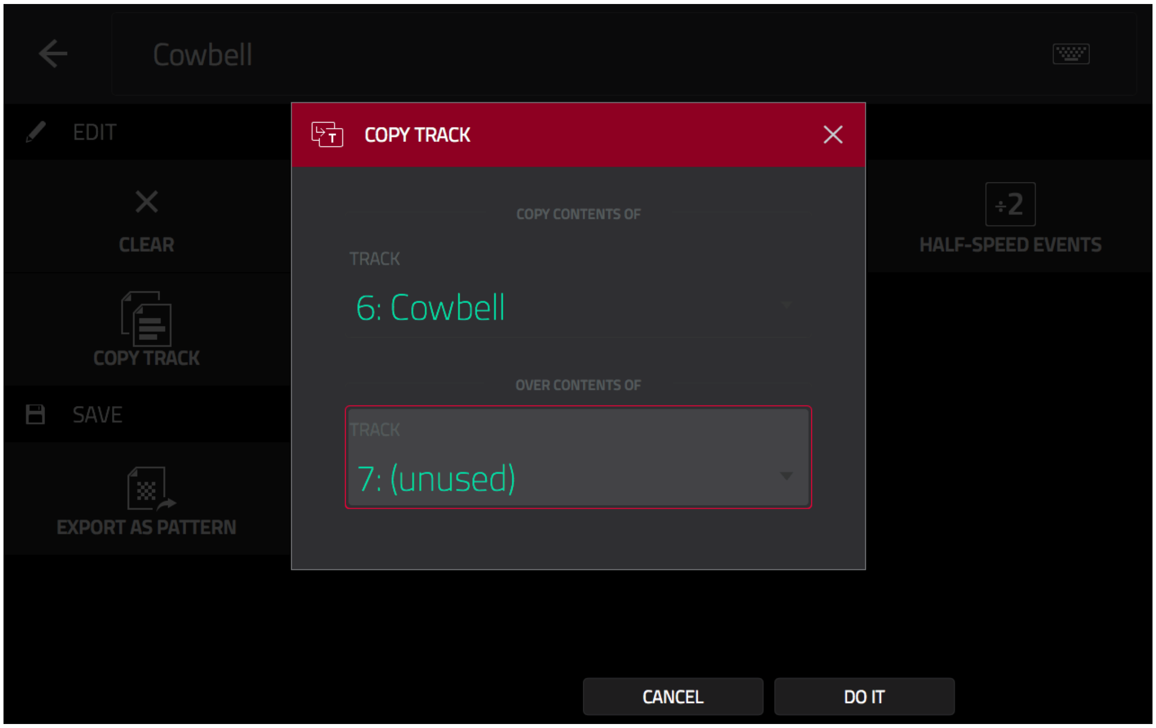
Using the Timestretched Loop

We can of course easily compare our tuned loop to the timestretched loop. Assign **the old-cowbell-timestretched** sample to pad **A02** in our **Cowbell Loops** program – again, set this pad to **NOTE ON**.

With the existing **Cowbell track** selected in **MAIN**, tap the **pencil icon** at the end of the **TRACK** row:

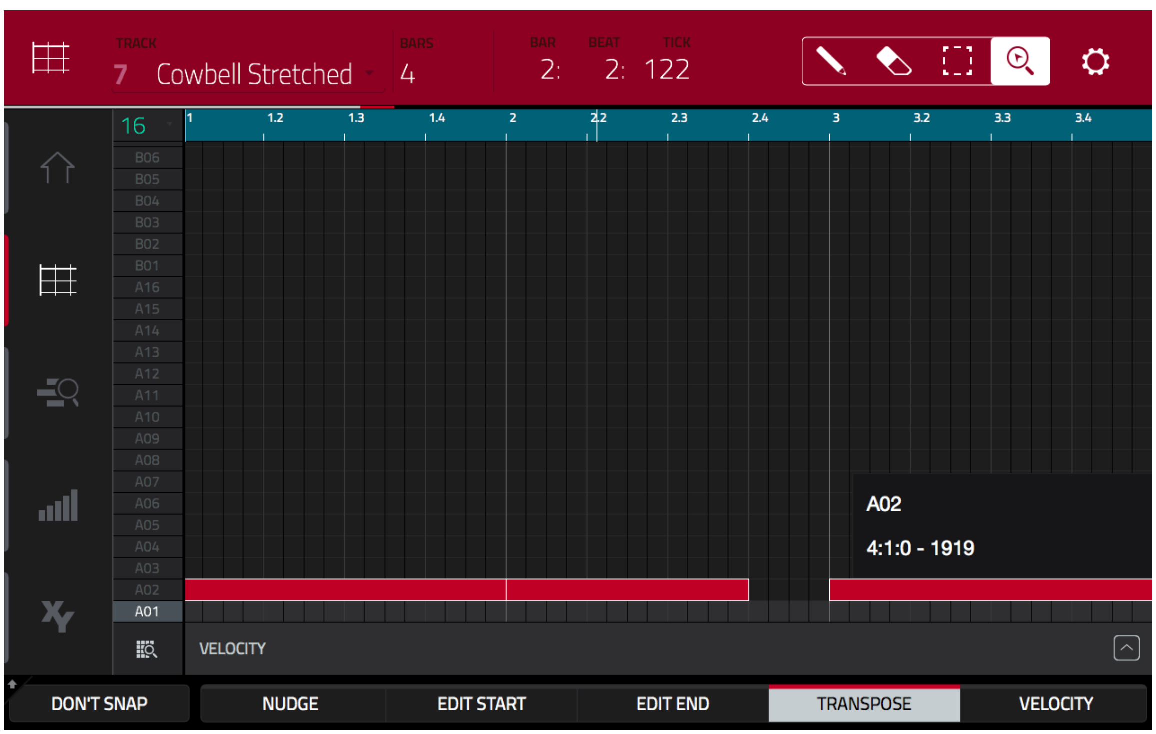


Select **COPY TRACK**:



This will make a duplicate copy of track 6 to track 7 of our sequence. **MUTE track 6** and **rename track 7** to **Cowbell Stretched**.

In **GRID EDIT**, select all the pad **A01** events for **track 7**. Hit **TRANSPOSE** and press the **+ button** to move them to **pad A02**:



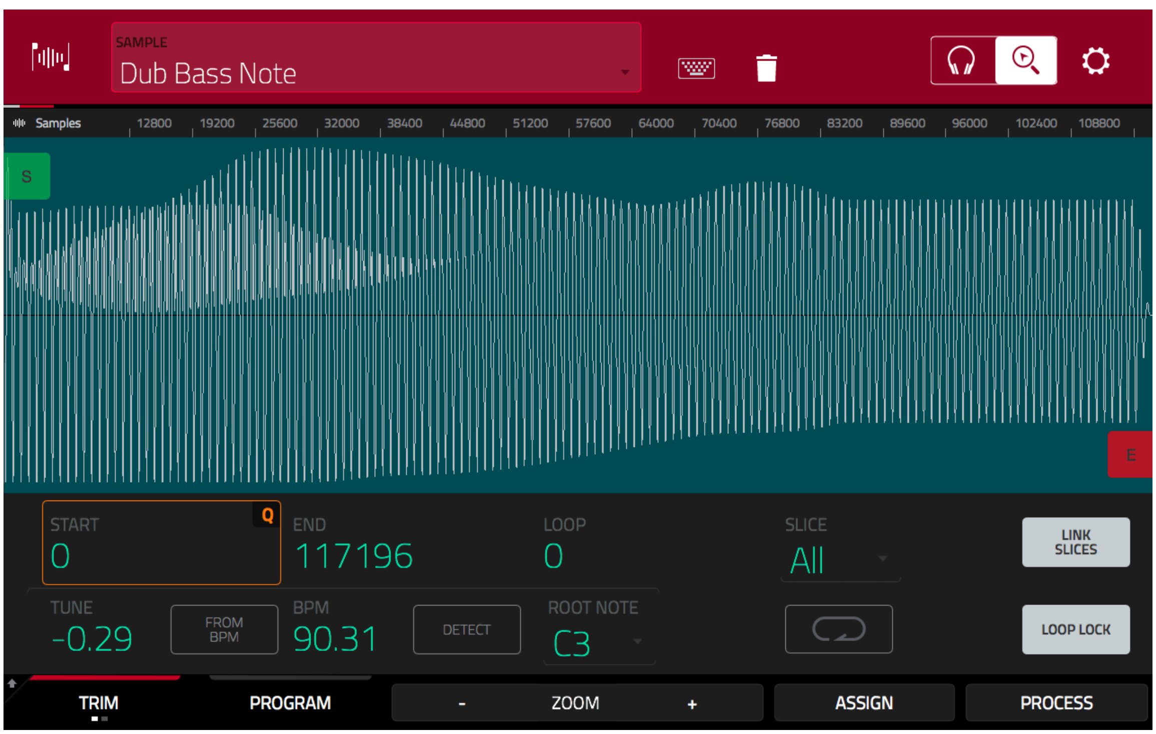
Press **PLAY START** to hear the timestretched cowbell loop in action – you can hear my version in **sequence 3** of the **Loops.xpj** project.

Switch between sequences 2 and 3 to compare how the sequence sounds with each version of the loop. Timing-wise they sound identical – the only difference is the tuned version sounds higher pitched. So both tempo adjustments worked well here, so you can choose either.

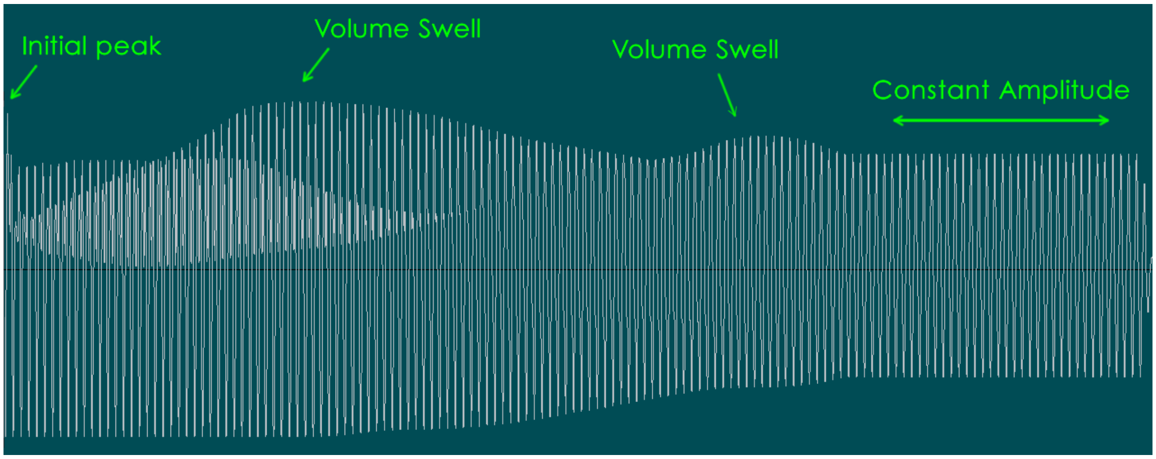
Looping Instrument Samples

Back in the keygroups tutorial, you might have noticed that the piano samples used were actually 'looped' and can be infinitely sustained by holding down the pad. Looping instrument samples however requires a slightly different looping approach; the process is often referred to as 'sustain looping'.

Let's sustain loop the **Dub Bass Note** we've been using in our **Shadow Kit** program. From the **'Sustain Looping'** subfolder**,** load up the '**Dub Bass Note'** sample and select it in **SAMPLE EDIT**.

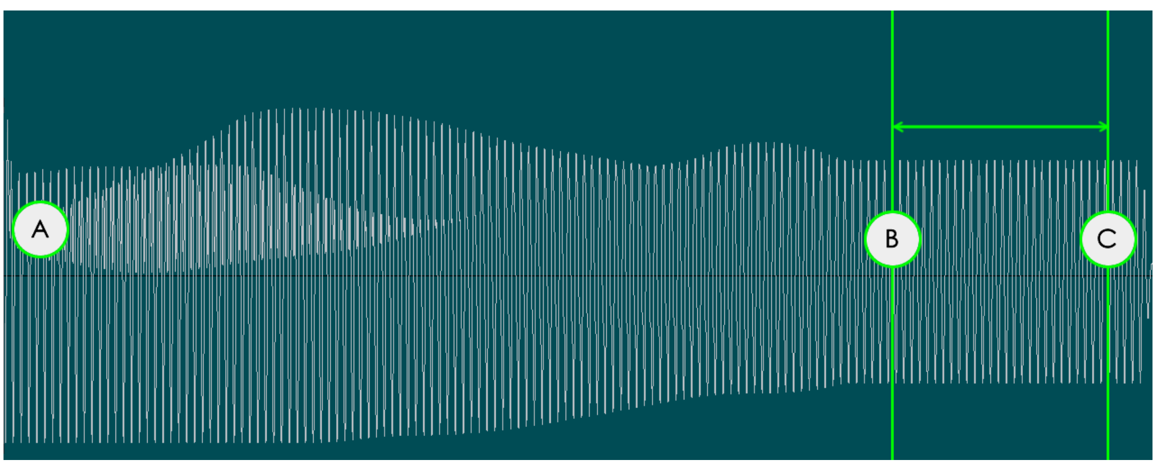


Now, looping an instrument sample requires a slightly different approach to 'standard' drum looping. First, hit **pad 16** to hear how our entire bass sample sounds when played back 'looped'. Hmmm, this is hardly a smoothly evolving loop, so clearly this is not currently achieving the desired effect – take another look at the waveform:



The waveform has a number of peaks, with an initial 'click' at the very beginning (the initial attack part of the bass note – it's mostly covered by the green 'START' point flag), then a gradual volume swell, then a smaller swell, and then a period of fairly consistent amplitude towards the end of the waveform. Basically there's no way this is ever going to loop 'seamlessly'.

What we want is for our sample to play its natural 'start' (i.e. the initial attack portion, which has a distinctive and unique sound), let it play through the 'swells' and then have it loop within the area where the amplitude remains fairly consistent:

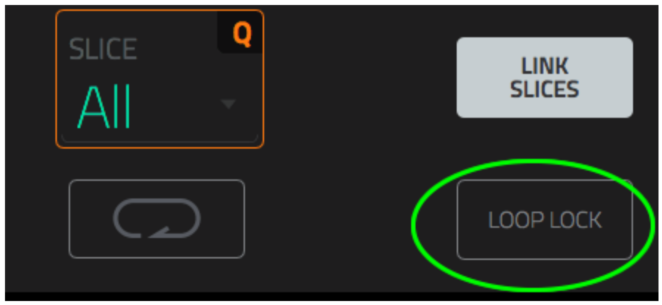


So in the diagram above, when we trigger the sample and continually hold down the pad, we get the initial 'attack' sound at point **A (Start point)**, the sample then plays to point **C** (**the end point**), and then jumps back to the **Loop Point** (**B**) and continues to loop between B and C until we let go of the pad. Unlike a drum or percussion loop, the **Loop point (B)** is not at the very start of the sample; it will have a very different value to the START point (A).

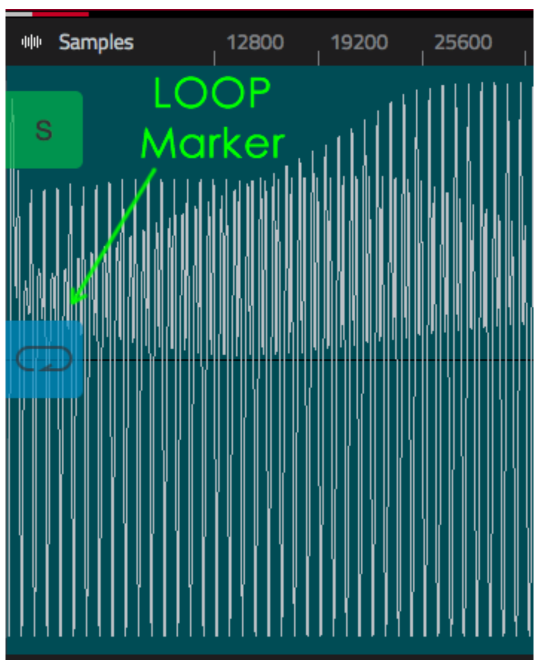
The looped region is clearly a region of (fairly) constant amplitude, and hence when the sample loops back, it should hopefully not be obvious that the sample is actually jumping back and forth between two different points in the waveform.

Setting the Looping Region

To set a sustained looping region, we must first 'unlock' the '**Loop point**' so it can be set independently of the START point. To do this, simply change '**LOOP LOCK**' to **OFF** by pressing the **LOOP LOCK** button so it is black, not white:



At this point you'll see a blue loop icon appear on your waveform – this is the **LOOP** point marker:



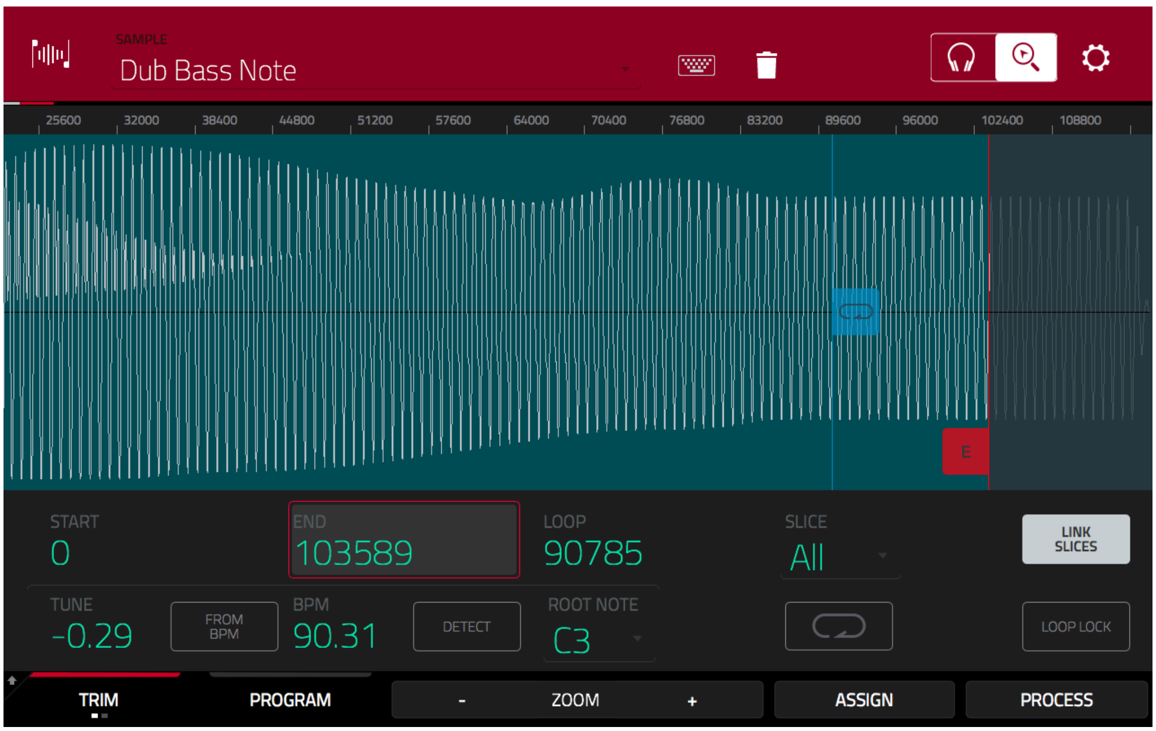
hold down **SHIFT** and set **SNAP 0** 'on'. Now press **pad 16** (CONTINUOUS LOOP) so our sample begins to loop back continually.

You can move the loop point in the same way as you would the start and end points; you can touch and drag that blue marker on screen, you can tap the

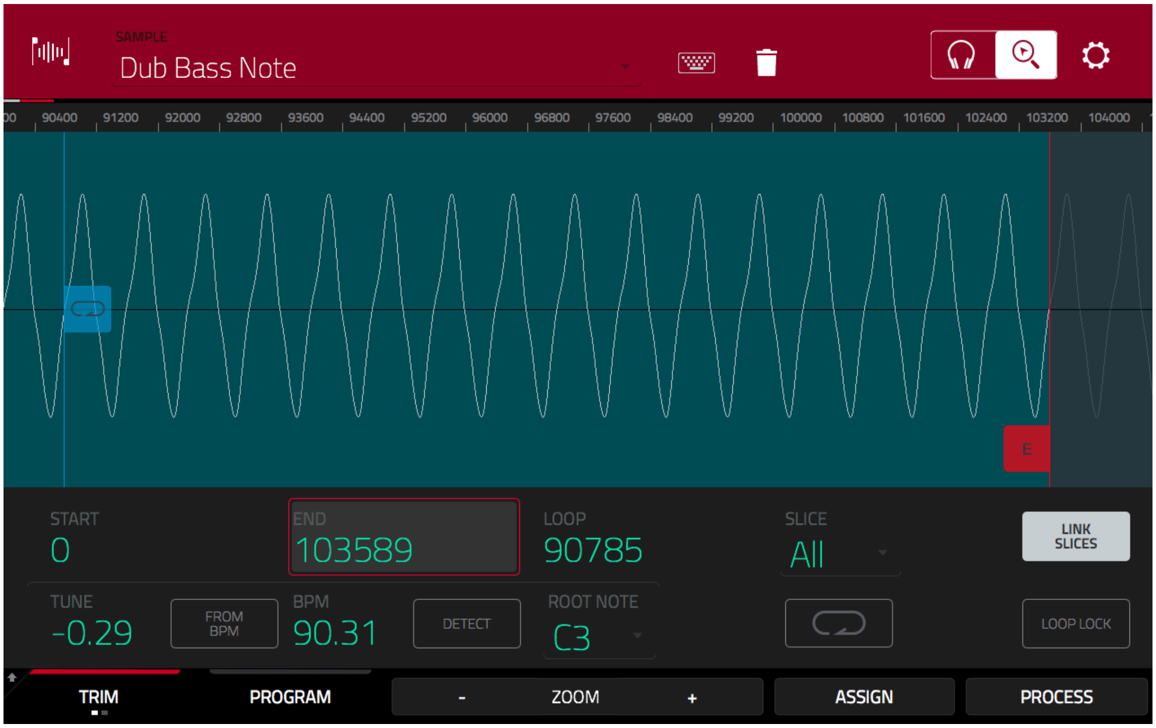
‘**LOOP**’ parameter and turn the data wheel, or you can use the Q Links; this time it’s the *third* bank of Q-Links that handle the independent loop point adjustment.

Set the **LOOP** point to around **90785** and adjust the **END** point so that our loop avoids the very end area of the sample (where it tails off very briefly) to one that also has a similar amplitude to that of the LOOP point.

Now as you have ‘continuous loop’ previewing active, you can hear the resulting loop in real time – as you adjust the END point, listen out until you (hopefully) hear a nice glitch-free loop. Try an **END** point of **103589**. Sounds great!



Zoom in to take a closer look:



As you can see, the SNAP 0 setting has worked its magic, ensuring both the LOOP and END point fall on an exact zero crossing. This loop sounds good because a) both points occur at the beginning of a waveform cycle and b) the waveform amplitude at both points is very similar.

Finally, we need to set the entire sample to **LOOP: Fwd** so it plays as a loop when used in a program.

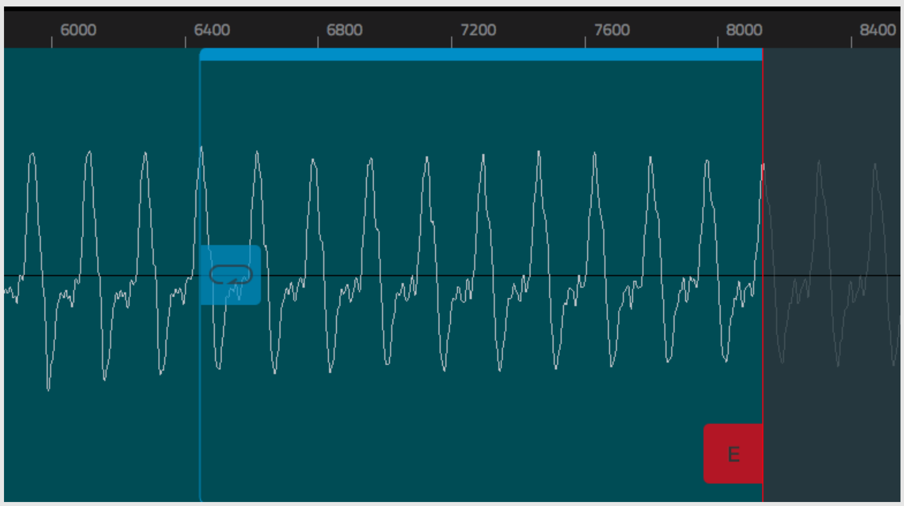
Press and hold down **pad 11** to hear how the entire sample plays through. As you can see and hear, the sample starts playback at the very beginning (the START point), plays all the way to the end point and then performs a continuous loop between the LOOP point and END point.

This sample can how be used 'as is' in the Shadow Kit program; you can now use it to create bass notes of any duration you desire. You can load up my version from the '**Sustained Loop**' subfolder, **Dub Bass Note-Looped.wav**.

Dealing with Problematic Loops

Not all sounds are this easy to sustain loop. I find synthesized sounds tend to be the easiest as they have very regular and consistent waveform cycles. Acoustic sounds have more erratic waveforms and you'll often find they require much more tinkering before you find a decent loop.

I've found that with some acoustic sounds I get better results by first turning **SNAP 0** 'off' and instead of looping between cycle crossing points, I instead loop between cycle mid-points:



Whatever you do, always make sure you 'mirror' the LOOP point waveform shape with the END point shape and amplitude – if you were to join them together they should produce a continuous waveform cycle.

Finally, it's normally best to choose the longest loop region possible, especially with acoustic sounds. However with synth sounds you can often get away with looping just a single waveform cycle.

015 Chopping Drum Breaks Part 1

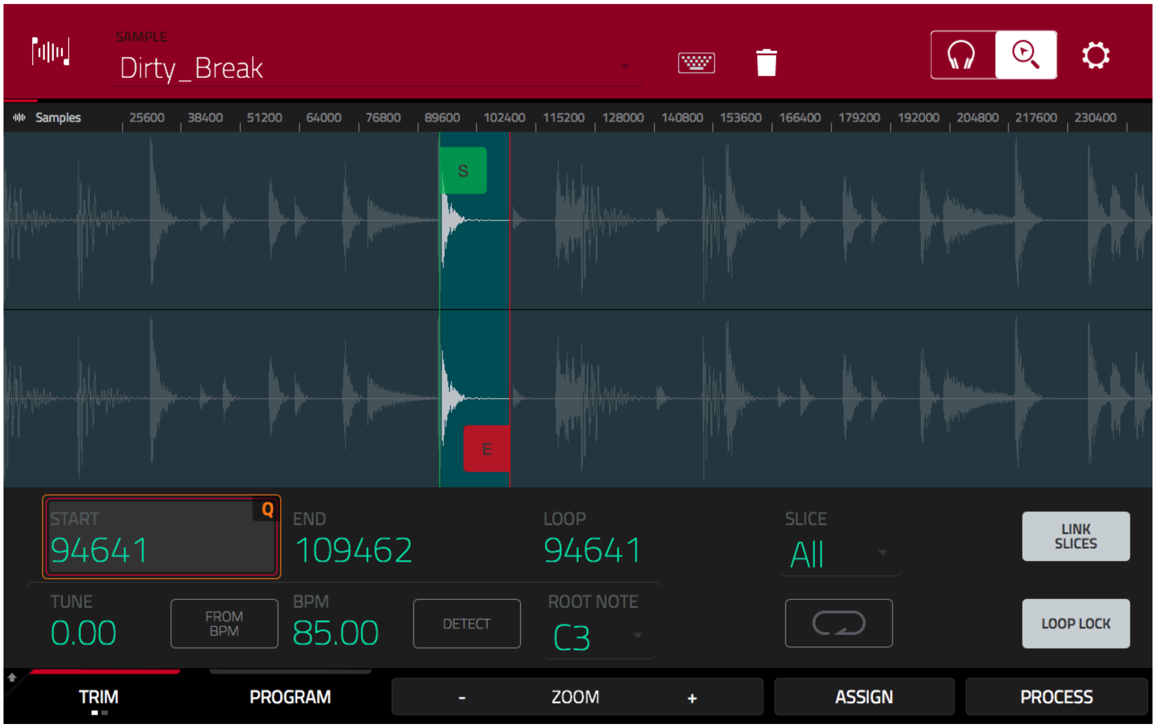
One of the most important skills a beat maker can possess is the ability to chop samples. Chopping is the gateway to a wealth of creative possibilities and the MPC offers perhaps the most intuitive and creative chopping functionality of any sampler or workstation!

Basic Chopping Using the Extract Function

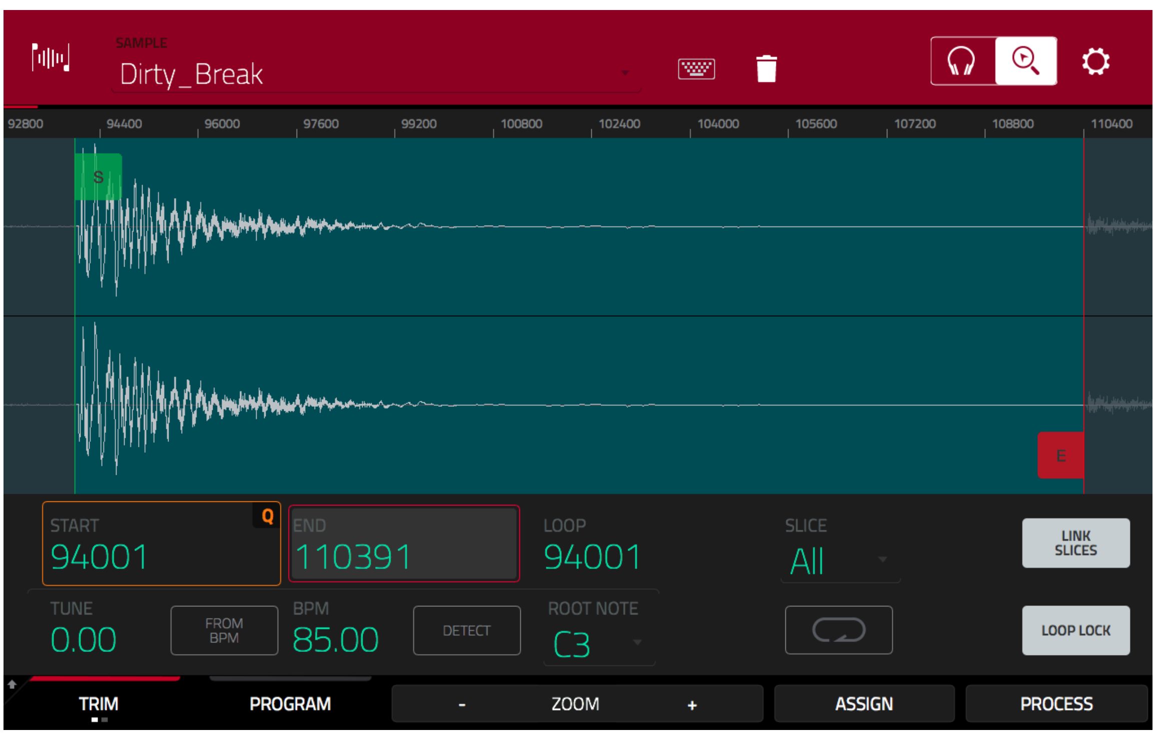
As we'll soon see, the MPC features a dedicated 'chop' mode designed for efficient and creative slicing ('chopping') of samples, loops and songs, however first let's look at the most basic way to chop a sample – using the '**Extract**' function.

Load the '**Dirty Break**' file from the **Chapter 015** folder and in **SAMPLE EDIT** have a listen by previewing the sample with the '**Play Sample**' pad **10**. It's a looped drum break, with a tempo of **85.0 BPM**.

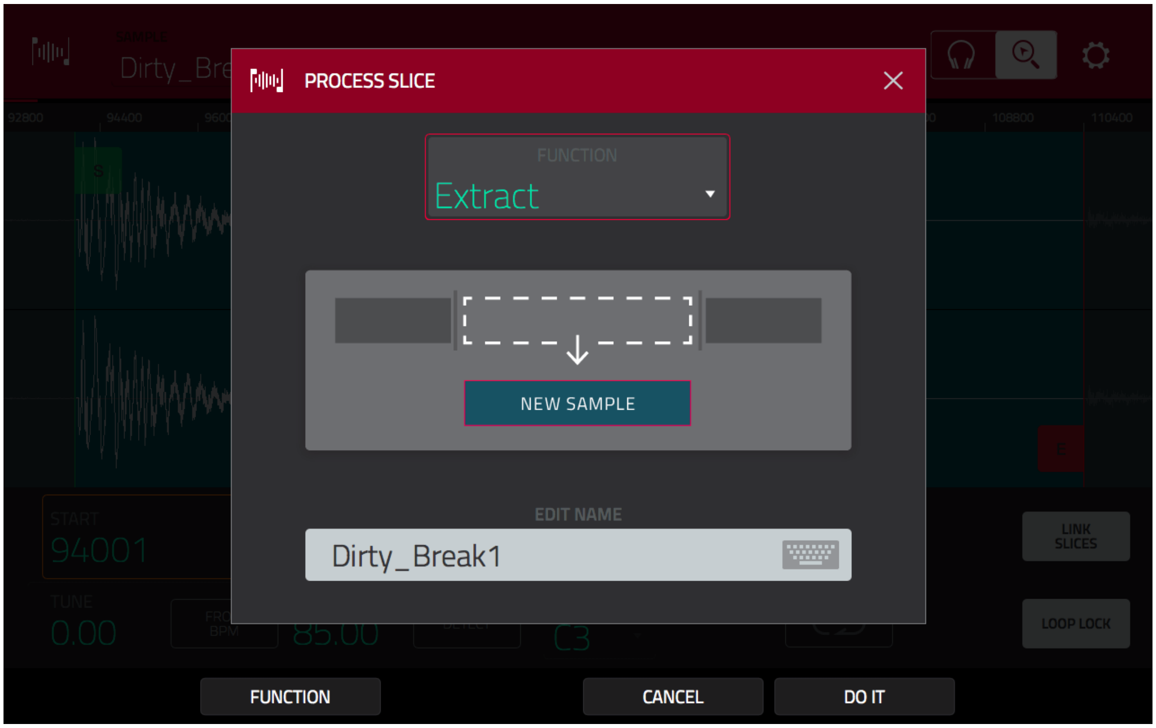
Let's say I just want to 'chop' that second main snare out of the break so I can use it as a 'standalone' snare sample in a drum kit. To chop it using the '**Extract**' function we'll first have to set our sample's start and end points so that they isolate nothing but that second snare. With **SNAP** 0 engaged, set your initial rough **START** point to **94641** and your **END** point to **109462**:



Zoom in and make your final adjustments – I set a **START** of **94001** and an **END** of **110391**.

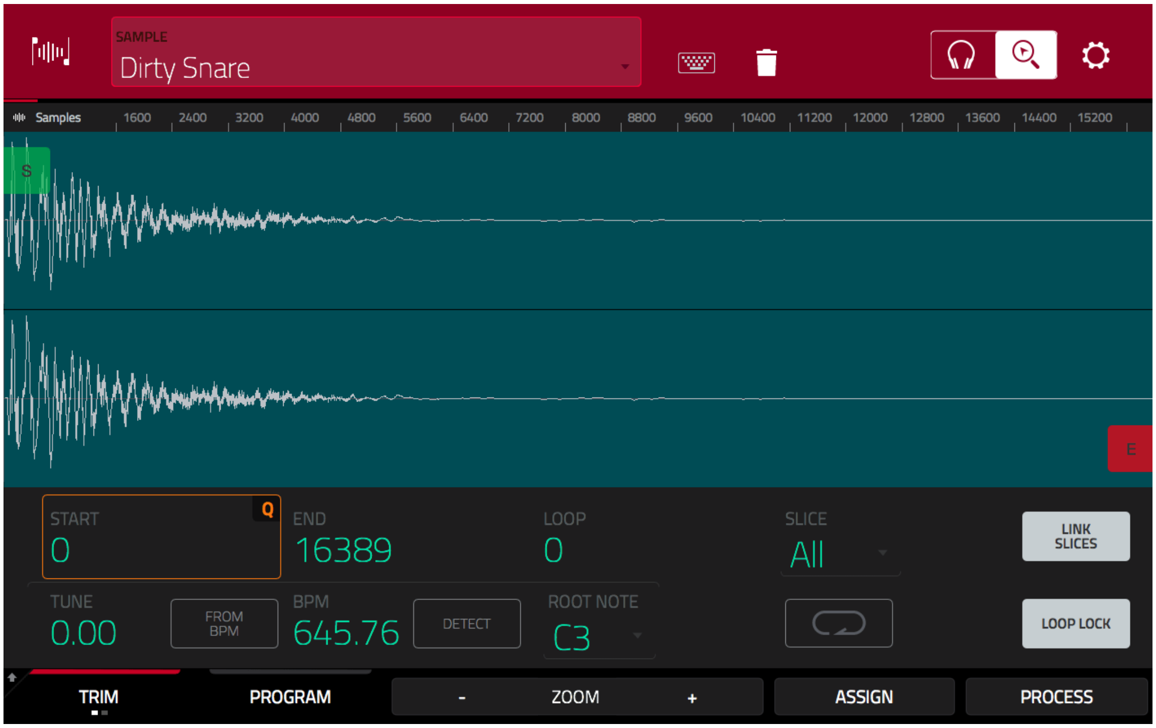


Press **PROCESS** at the bottom of the screen and turn the data wheel clockwise to select the **EXTRACT** function:



This function is the opposite of DISCARD as it takes all the data *between* your edit points and creates a new sample out of them. You'll see the MPC has suggested a new name for your extracted sample, '**Dirty_Break1**' – tap in the text field and rename it **'Dirty Snare'.**

Pressing **DO IT** creates the new 'Dirty Snare' sample but also leaves your original break completely intact (the start and end point markers will still currently be set either side of the snare). So using this method you now have two samples – the original break and the extracted snare. To view the new snare, tap on the **SAMPLE** name field in the top left of the screen and use your data wheel to scroll to the 'Dirty Snare' sample (or double tap to select it from the pop up list).

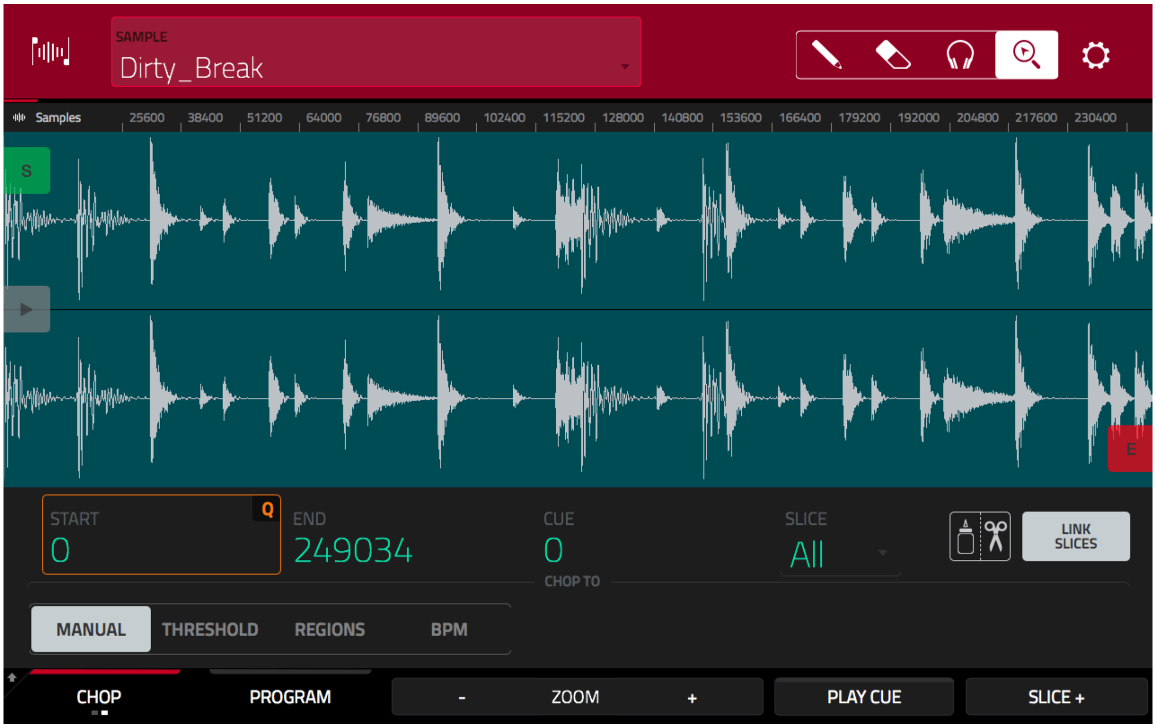


Using the Dedicated CHOP Mode

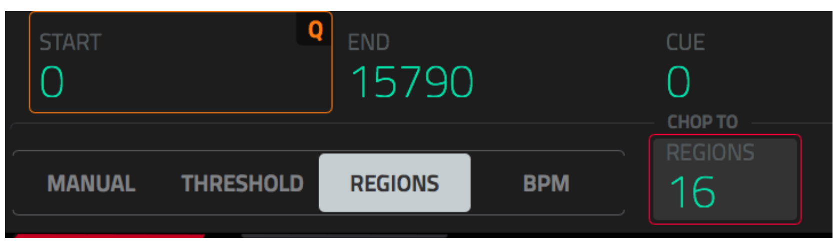
The EXTRACT function is a perfectly reasonable way to extract individual portions of your samples, however there is a far more efficient way to 'extract' multiple sounds or regions from a break; **CHOP** mode.

Let's start with the original '**Dirty Break'** file before we changed the start/end point – you could just reload it from the BROWSER, use UNDO history, or just drag the start and end point markers back to their respective ends.

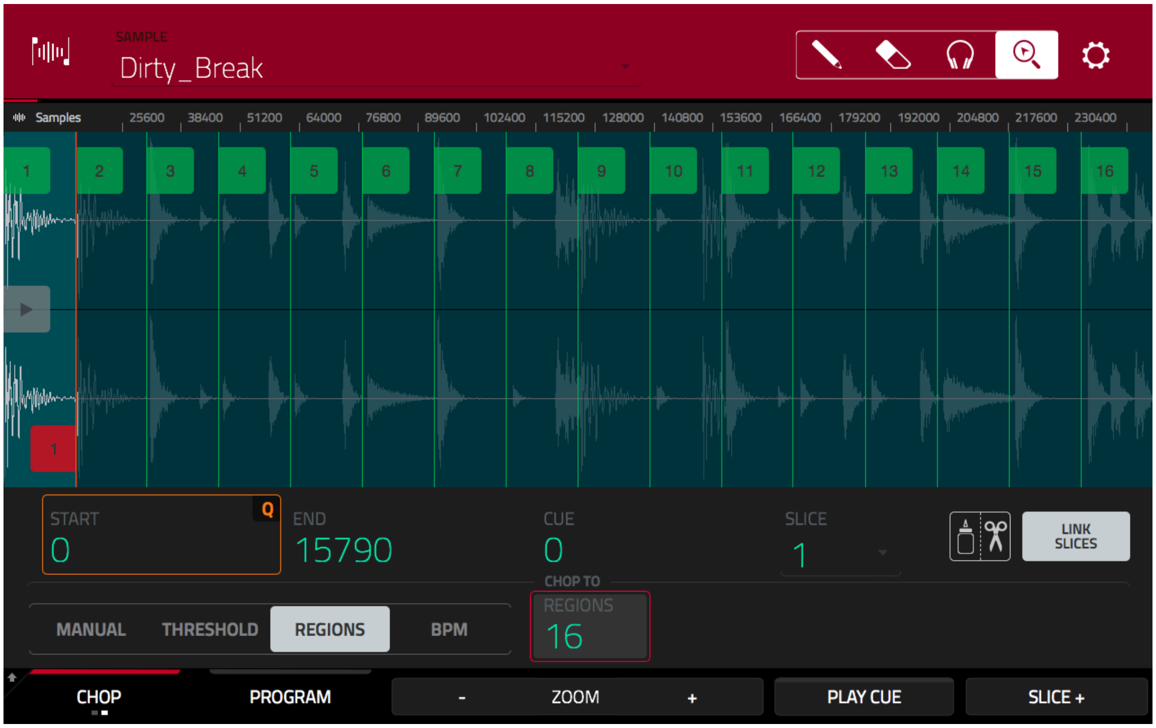
In the **SAMPLE EDIT** screen press the **TRIM** button in the bottom left of the screen so the button changes to read **CHOP**. You'll now be in **CHOP MODE**:



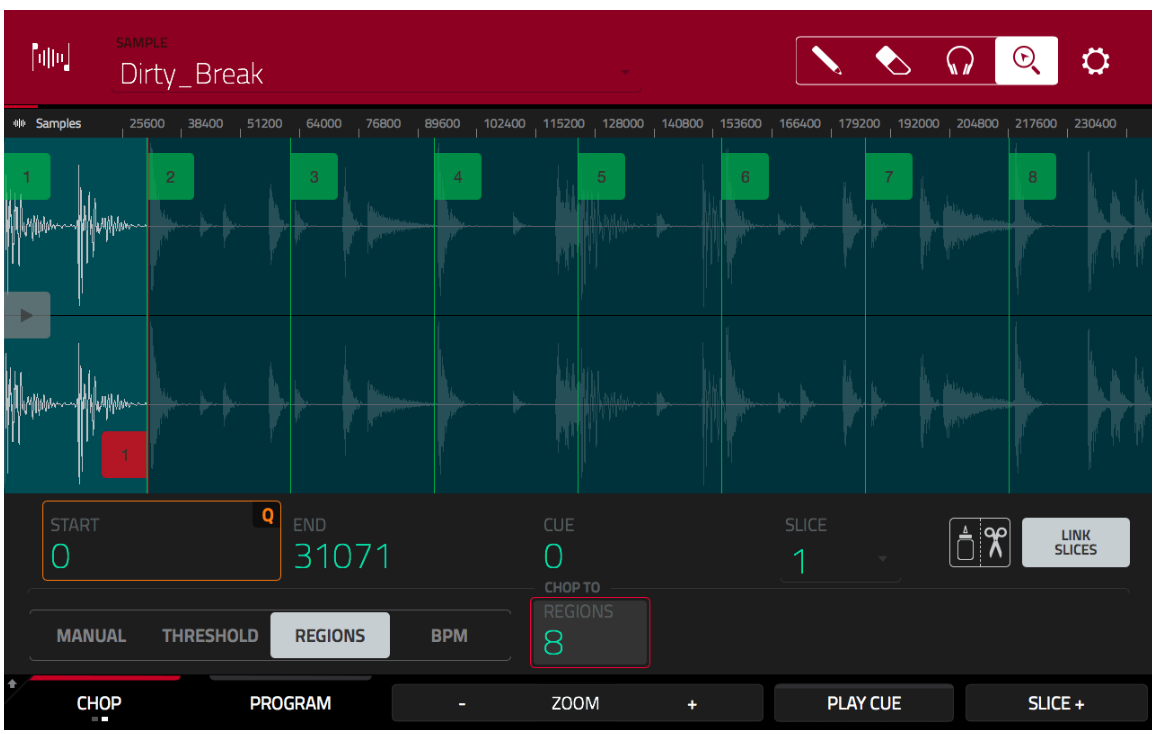
Locate the '**CHOP TO'** options panel at the bottom of the screen – select the **REGIONS** button:



When chopping to 'regions', the MPC initially splits your break into 16 equal parts:



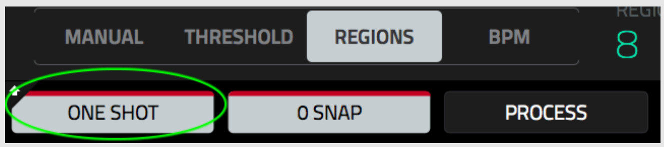
Tap **REGIONS** and change this from **16** to **8** – the MPC will now be splitting your break into 8 equal parts:



Notice the eight illuminated pads? Each region is now automatically assigned its own 'preview' pad, with the currently selected region in green, the rest yellow. Hit pads **A01** through to **A08** to hear each individual region.

Pad Preview: ONE SHOT or NOTE ON?

Hold down **SHIFT** and you'll see the **ONE SHOT** option at the bottom left of the screen:



When set to **ONE SHOT**, each pad hit will preview the region as a ONE SHOT sample. Tap this again to change the preview to a **NOTE ON** style preview – now the region stops playing when you let go of the pad (handy when you are editing the start point of a long region).

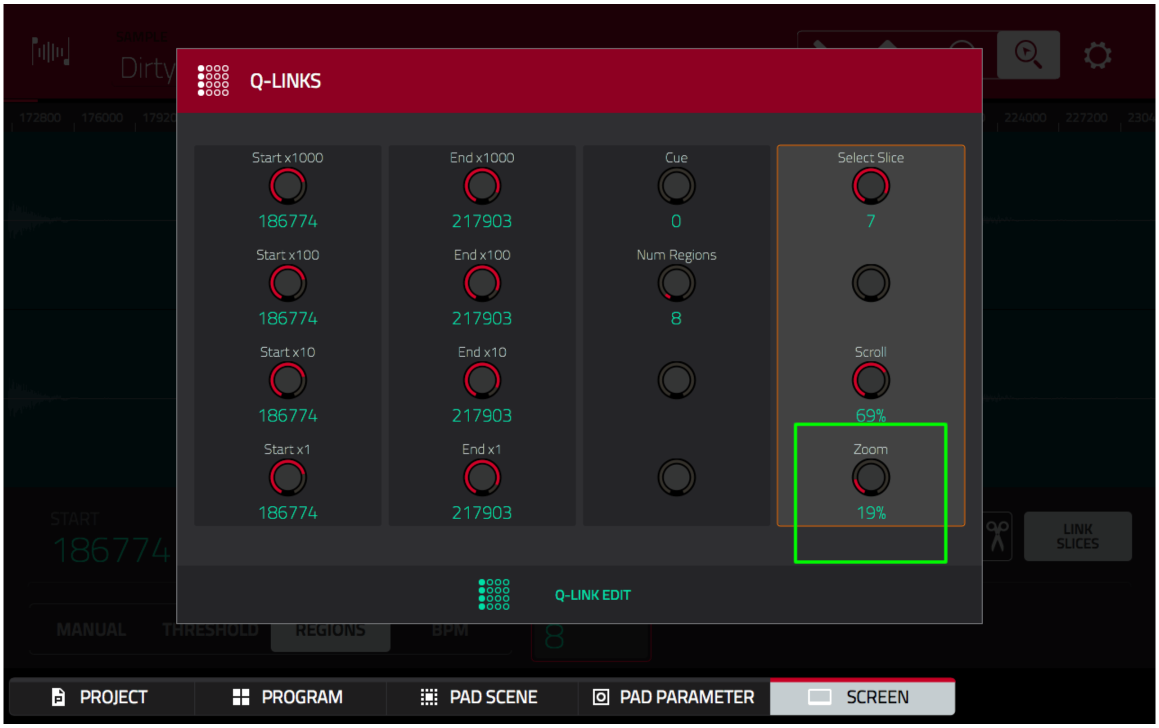
Identifying 'Problem' Regions

It is worth noting that a REGIONS chop is not performing any kind of waveform transient detection – it just divides your entire waveform into **equally sized regions** irrespective of the actual audio content itself. As such, it is often the case that you'll find that some regions are not perfectly lined up with the transients in your break, especially when dealing with live breaks performed by an actual drummer (they don't tend to have quantise options!).

The first thing I would say is that you don't always have to 'fix' regions, because often the regions still 'sound' fine. However when region chopping there are often three problems you may come across that may need some attention.

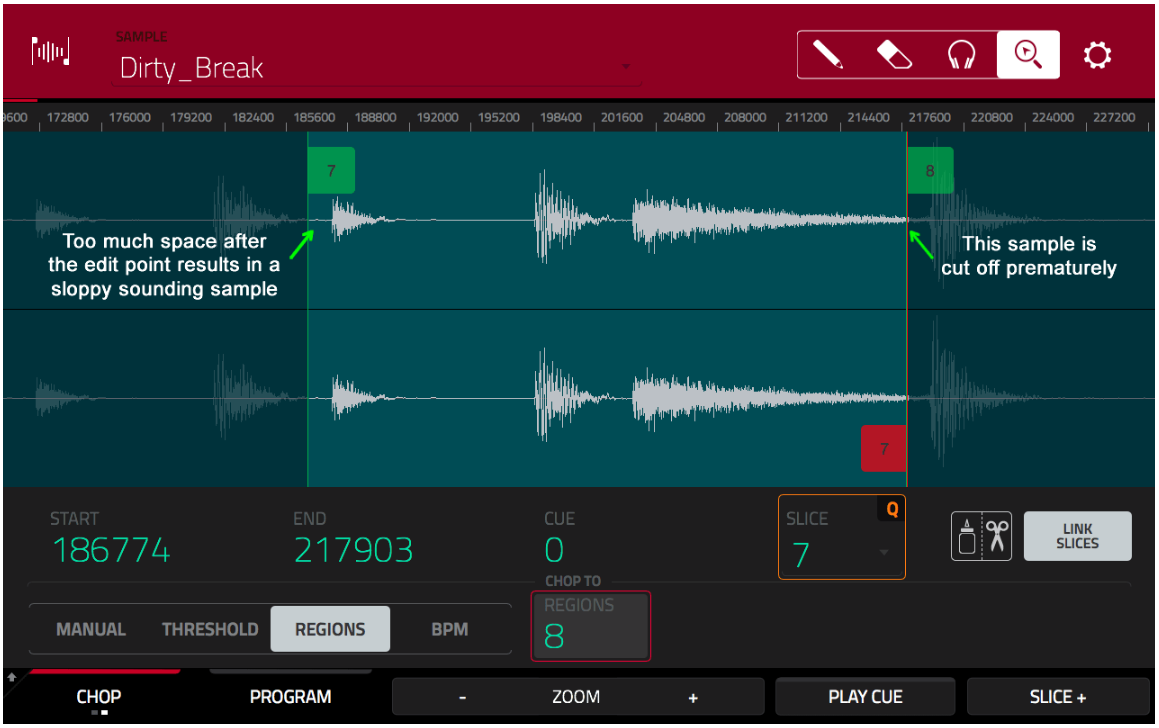
Firstly some regions may have a nasty click at the end (we'll look at fixing this in detail in the next chapter). Secondly some regions will be too short with too much audio prematurely discarded at the end of the region. And thirdly some regions just have a bit too much space at the *beginning* of the region which means the resulting slice sounds a bit delayed and not very 'tight'. Let's illustrate this with an example.

First, set a **Zoom level** of around **19%**. You can view the current zoom level by holding down the **Q-LINK** button - the zoom dial is **Q-LINK 4** (this is the bottom Q-LINK, column 4):



To magnify the waveform you have three options; you can 'pinch and zoom' using the magnifier tool (as described in chapter 8), you can turn Q-LINK 4 clockwise or you can press the **Zoom In** (**F4**) button - a zoom level of 19% is achieved from pressing **Zoom In** two times.

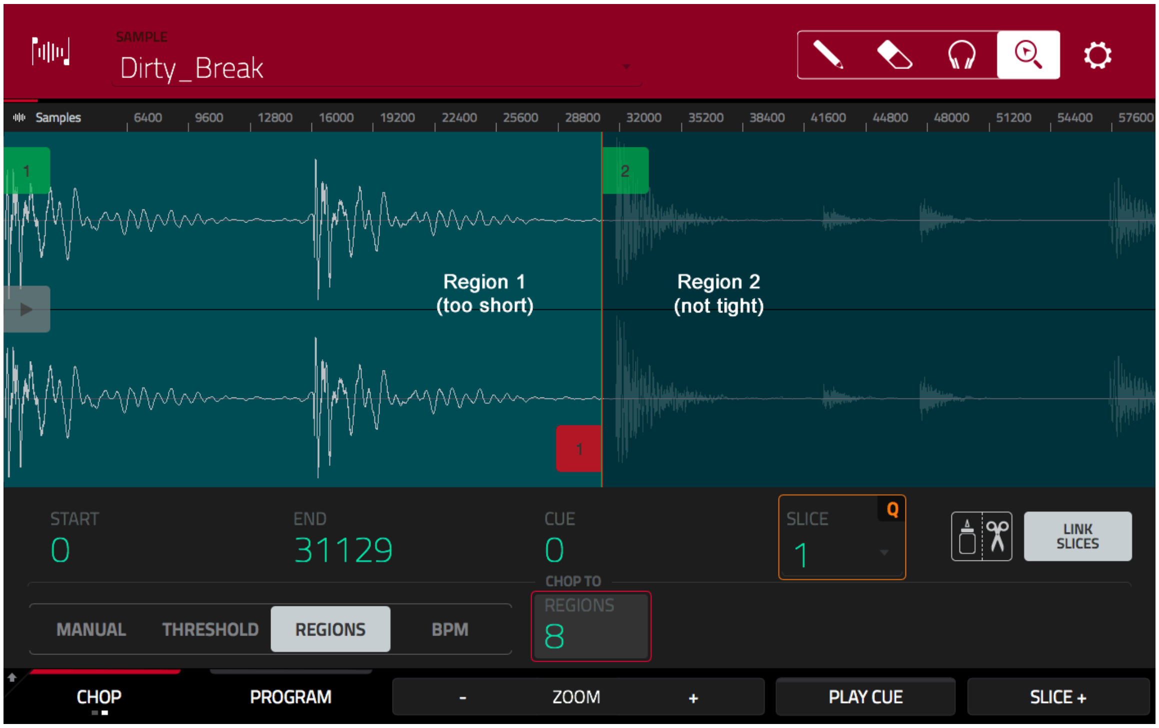
After setting a zoom level of **19%**, hit **pad A07**.



As you can see, the beginning of the region is not very ‘tight’ at all. If you hit **A07** to preview the slice there’s definitely an audible delay before that hi hat comes in, it just sounds sloppy.

Additionally, at the end of the region you can see that the open hat is cut off prematurely - ideally we want to salvage as much of this open hat as possible.

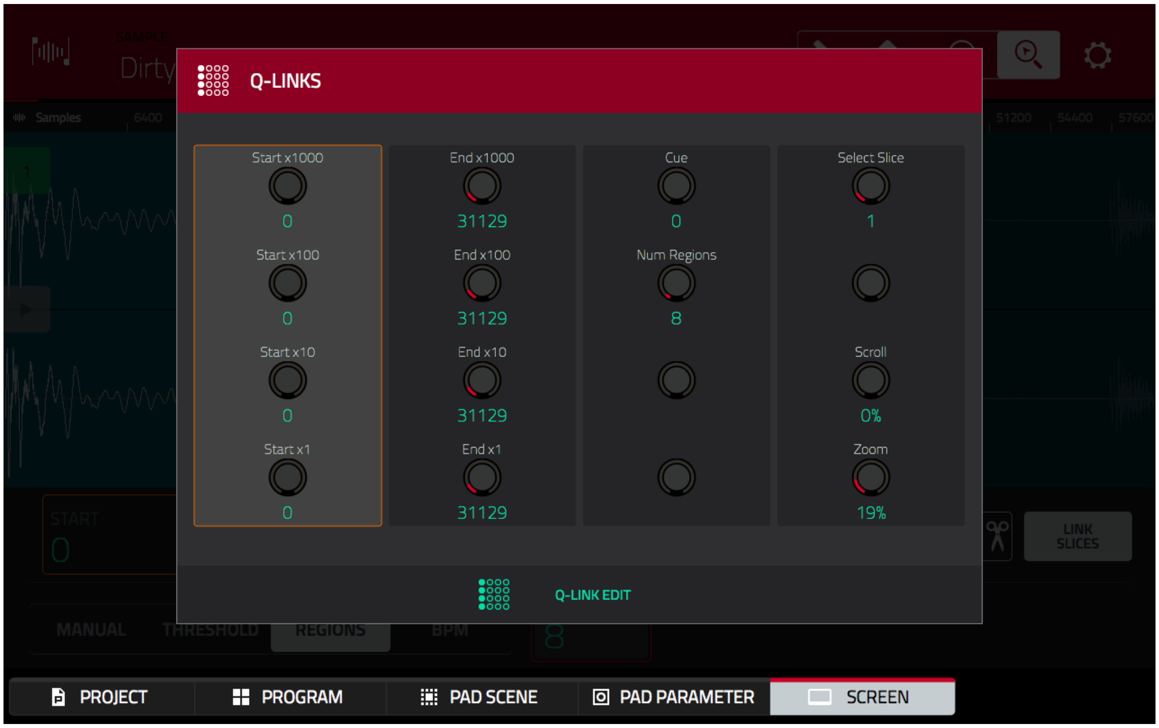
Take a look at all the regions and you’ll see that this problem is present in all of them. For example, here’s region 1, which is cut off prematurely - you can also see that region 2 is not ‘tight’.



Let's look at a nice speedy 'chopping' workflow to get these issued rectified very efficiently.

Fixing the Regions - Speed Chopping Workflow

We covered basic sample editing techniques in chapter 8 using SAMPLE EDIT 'TRIM' mode, and the core techniques are the same here. However, here in CHOP mode the Q-LINKS are configured a little differently to normal TRIM mode - check out the **Q-LINKS** panel in the **Panel Strip**:



The START and END point Q-LINKS are the the same, but this time there's no 'LOOP' Q-LINKS (there's no looping in CHOP mode).

This speed chopping workflow is designed to use an efficient combination of hardware and software. Give it a go and see how it works for you, just remember that there are so often several ways to perform the same task in the MPC so you can always substitute some techniques with your own favourites.

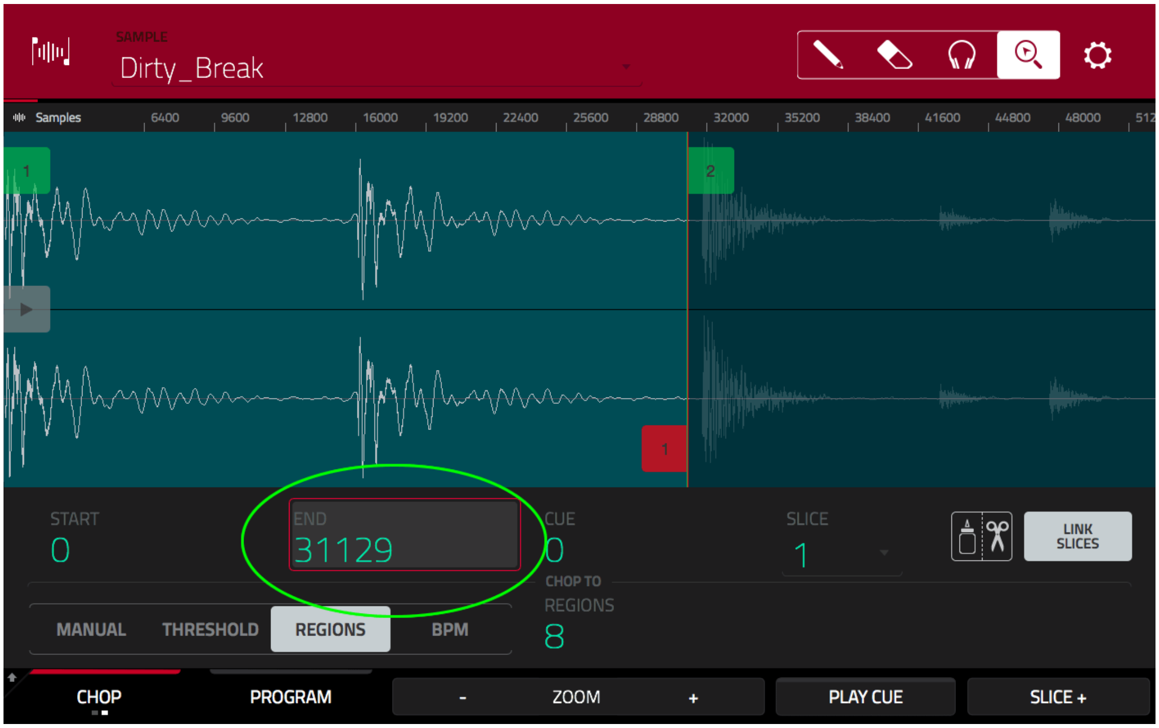
The first three steps are used to configure the workflow:

1. Set SNAP TO ZERO

As covered in chapter 8, setting SNAP TO ZERO ensures all our edits land on a nice 'click-free' zero crossing point. This is achieved by holding down **SHIFT** and pressing **0 SNAP**.

2. Select the END point

In this workflow we're only going to edit 'END' points (I'll explain why very soon), so in order to ensure the MPC always zooms into the end point of our region we'll need to 'select' the END point parameter in the hardware. So in your hardware, tap the '**End**' parameter:



3. Magnify the waveform

You'll need to magnify the waveform so you can see the edit point clearly. Try to pick a magnification level that still shows some of the surrounding waveform, a **Zoom level** of approximately **60%** works well, which correlates to hitting the **Zoom +** button **6 times.**

With the workflow configured it's time to begin editing.

4. Select the region to edit

Tap the pad relating to the region you wish to edit. In this case, **pad A01** for **region 1**.

5. Check the waveform

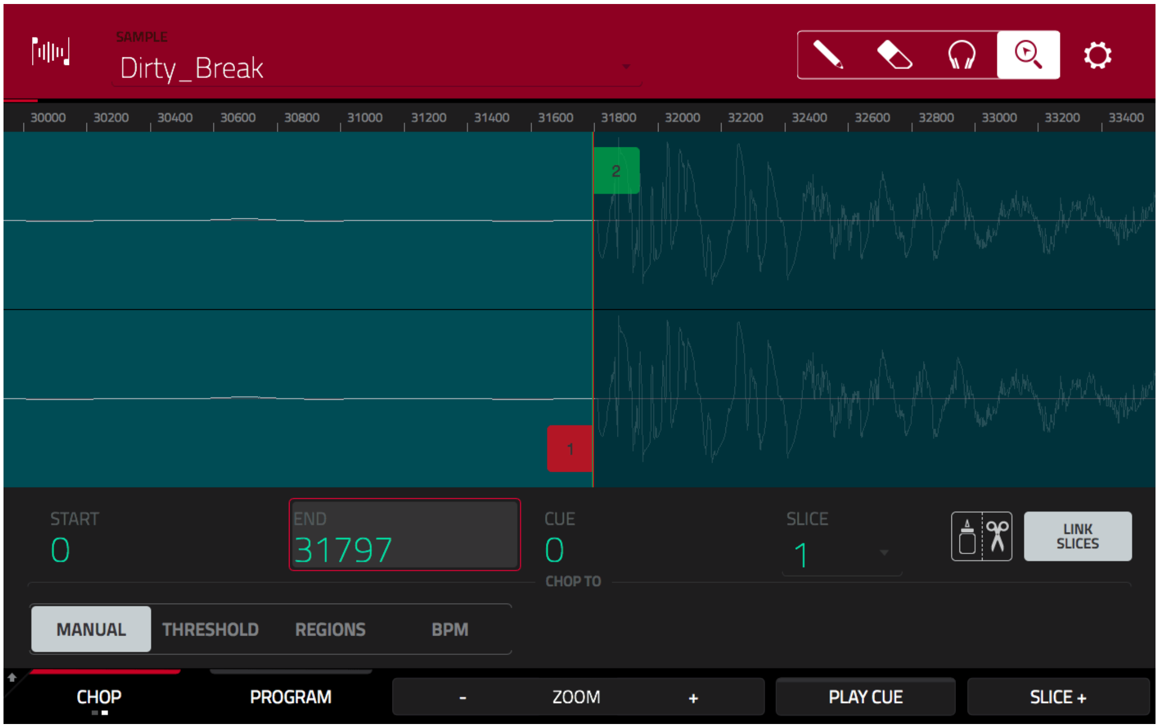
Quickly examine the waveform for any obvious problems. Also use your ears to listen out for any nasty clicks. As you can see from the above screenshot, the end of region 1 is clearly too short so we are going to need to extend the 'END' point a little.

6. Adjust the END point

We know from chapter 8 that there are many ways to adjust the end point in the MPC - for example you can drag the red 'end point' flag however in this workflow I'll be changing the END point using the hardware.

While you can use **Q-LINK 2** (bottom Q-LINK, column 2) to make fine END point adjustments, I prefer to use the data wheel. Remember we have SNAP TO ZERO set and also have a decent magnification so each turn of the data wheel should land us on a good edit point.

So turn the data wheel clockwise while looking at the waveform until the END point is taken right up to the beginning of the transient from the next region, in this case an **END** of **31797**:

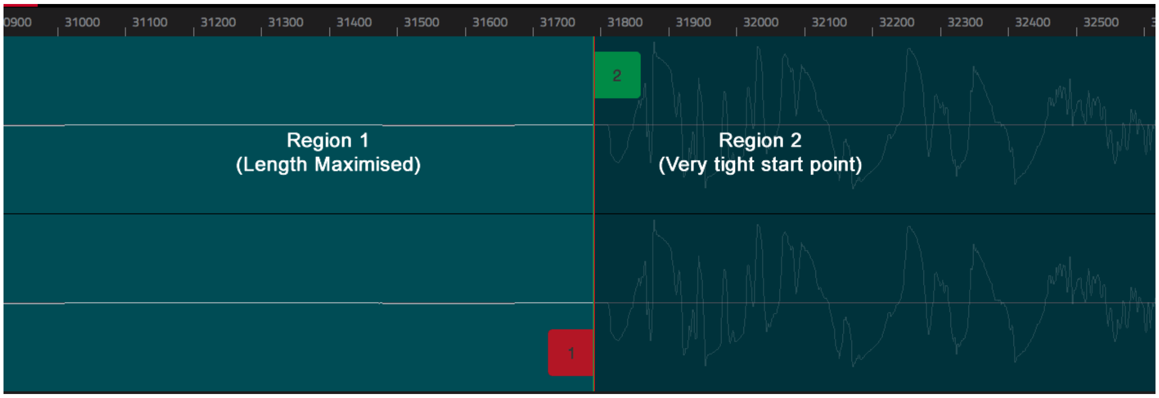


So in this workflow I use my right hand to turn the data wheel and my left hand to tap the region selection pads while view the waveform on my screen.

Once you are happy with the edit from a visual perspective, quickly tap pad A01 again to have a quick listen and if all sounds good repeat the workflow from step 4 onwards for all the other regions.

Shared Edit Points

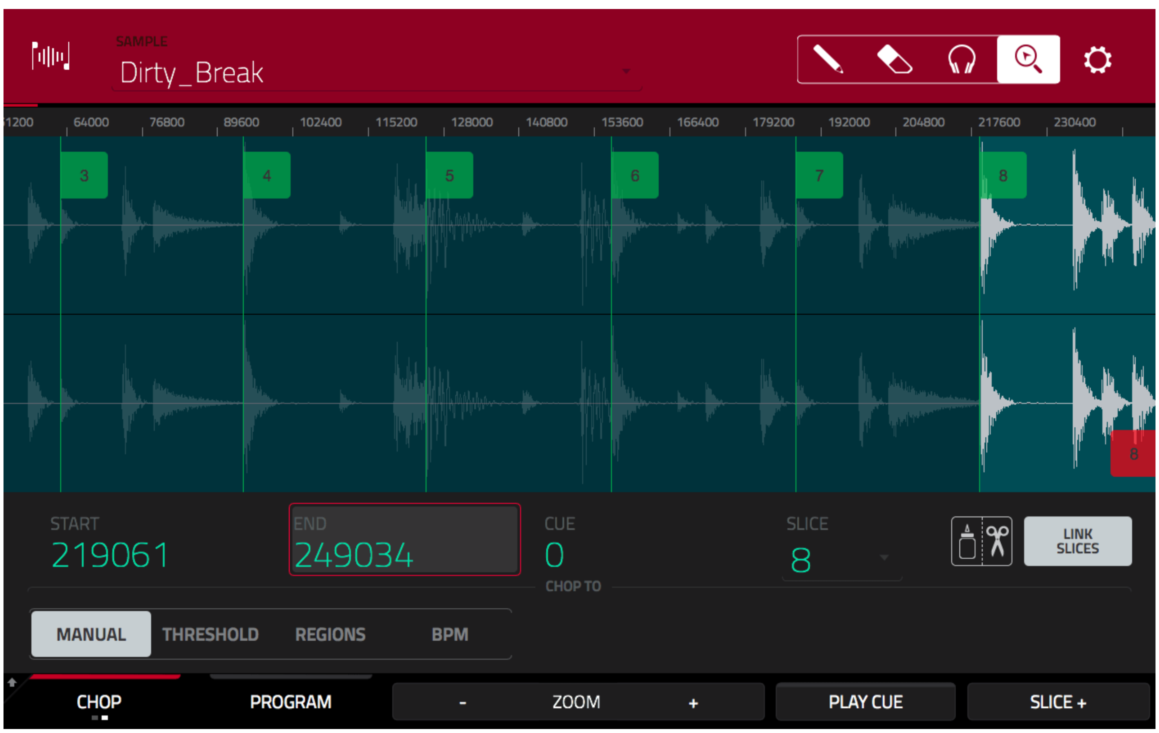
Notice how by adjusting the END point of region 1 we've also simultaneously adjusted the START point of the next region (region 2) to make it nice and tight.



This is because by default, adjacent chop regions have a 'shared' edit point; so the end point of region 1 has the same value as the start point of region 2. Get that one 'shared' edit point correct and we've taken care of two issues; the premature ending of region 1 and the sloppy start to region 2.

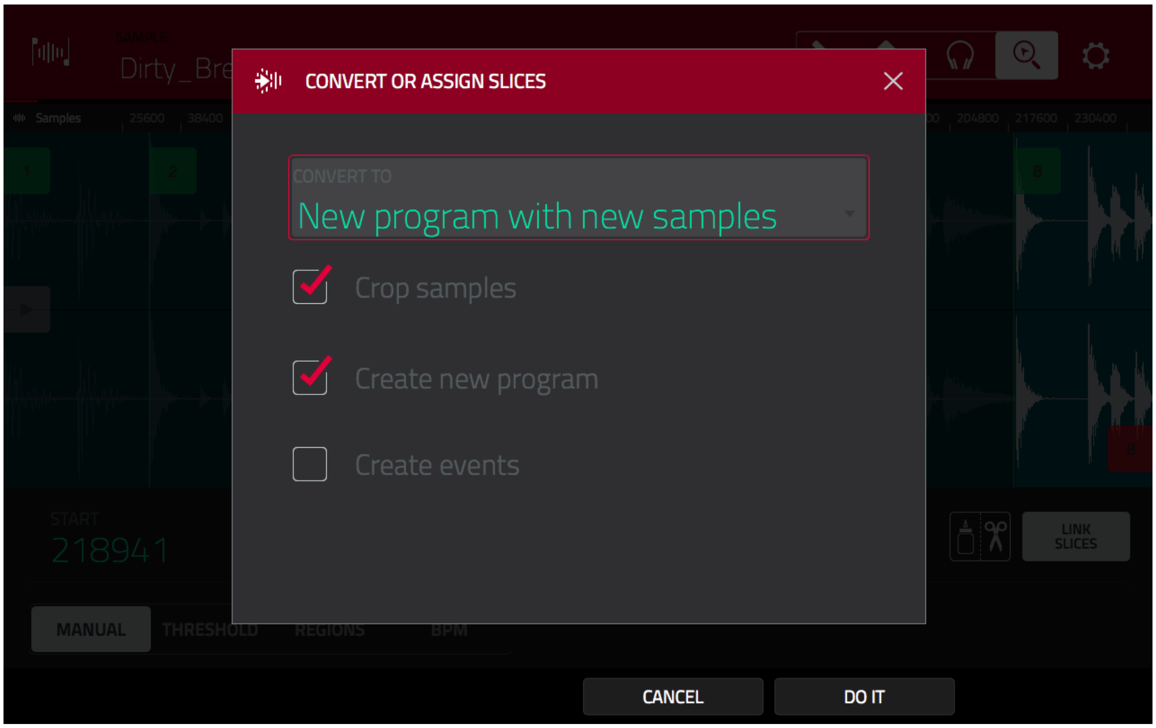
This is why we only need to edit the 'END' point in this workflow, as editing the END point of one region also fixes any issues with the start of the next region. This makes the editing process quick and painless.

Here's the final edit after adjusting all regions:



Exporting Your Regions

So how do we actually use these chop regions? Hold down **SHIFT** and hit **CONVERT**:

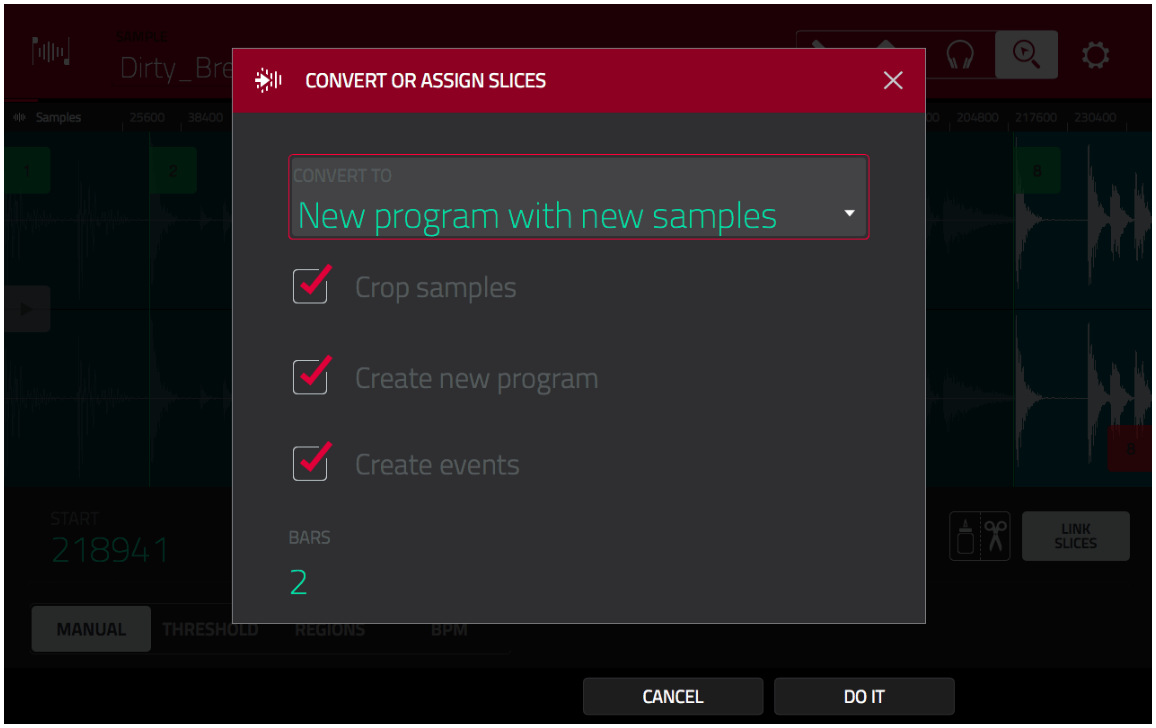


'**CONVERT TO**' gives us several different conversion types, but at this stage I want to pick the simplest type of conversion, '**New program with new samples**'. Upon selecting this, the options available change:

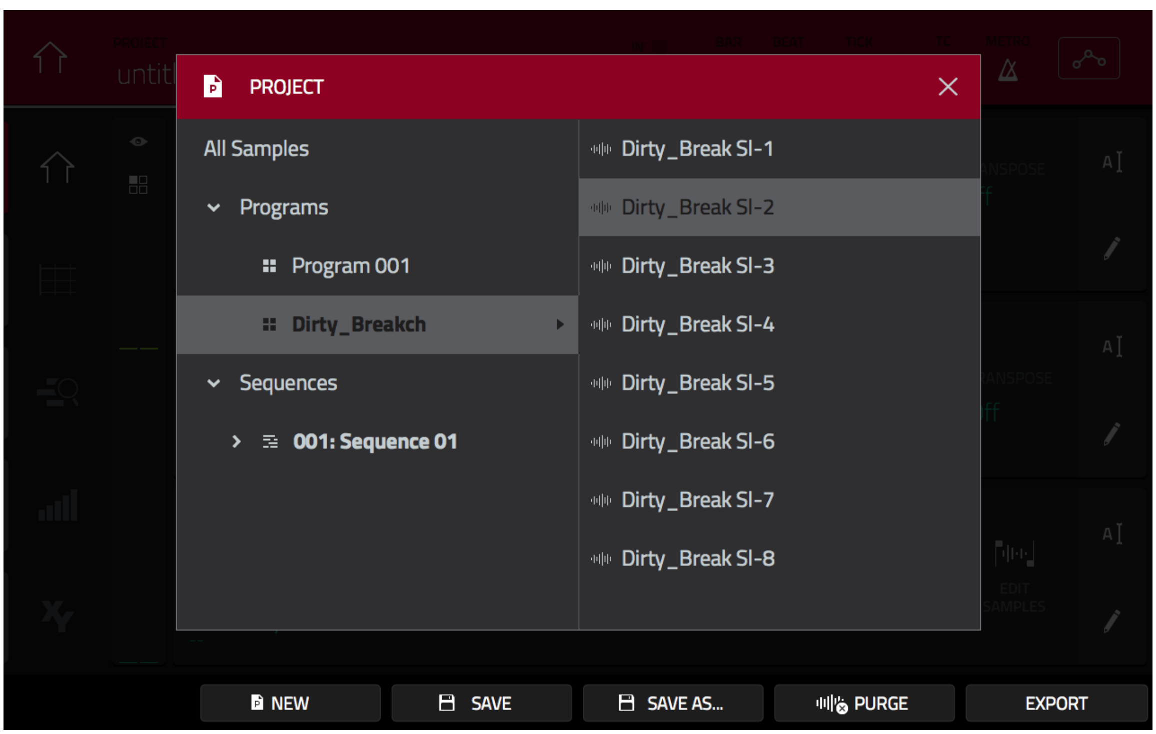
'**Crop Samples**' means the MPC will create a unique sample out of each region, so leave this checked.

'**Create new program**' will create a new DRUM program and automatically assign our new chopped samples to their own unique pad – leave this checked.

Finally, make sure you check **'Create Events'** – this is going to create a new sequence that when combined with our new program, will recreate our original break in the MPC sequencer. Upon checking 'Create Events' you'll be asked how many 'BARS' make up your break. Leave this to the default '2' (remember in 4/4 music, 1 bar is 4 beats, our break has 8 beats).

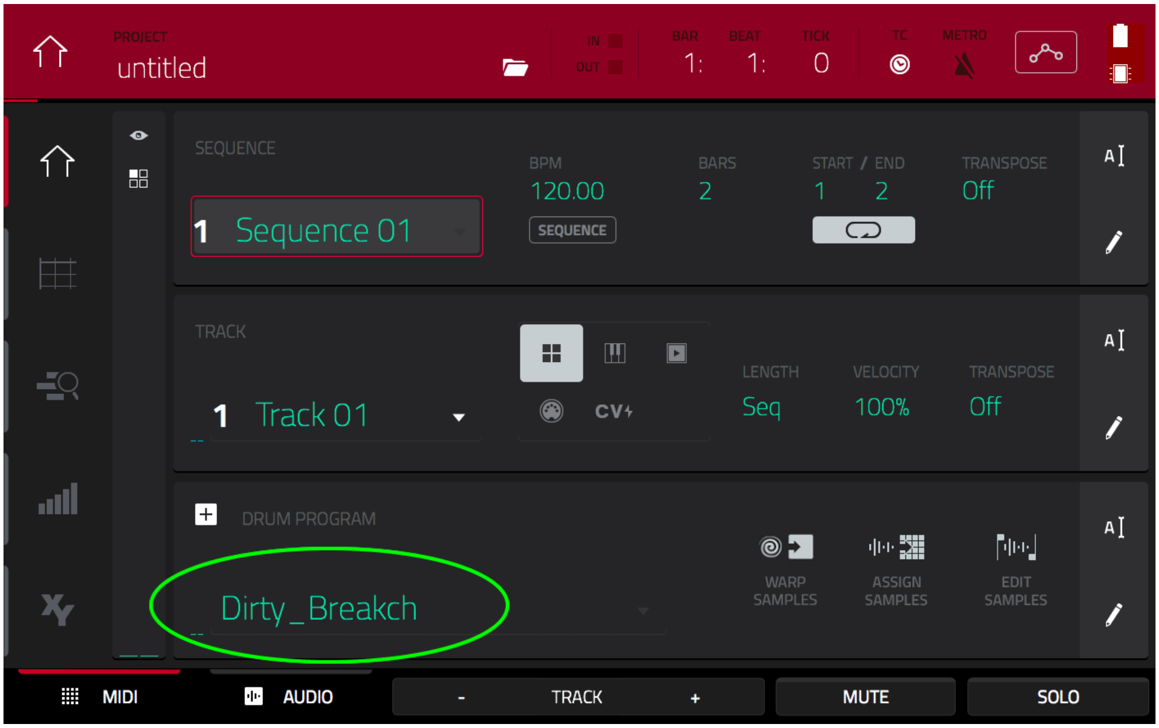


Hit **DO IT** to perform the conversion. Go to **MAIN** and press the **PROJECT** icon at the top of the page:



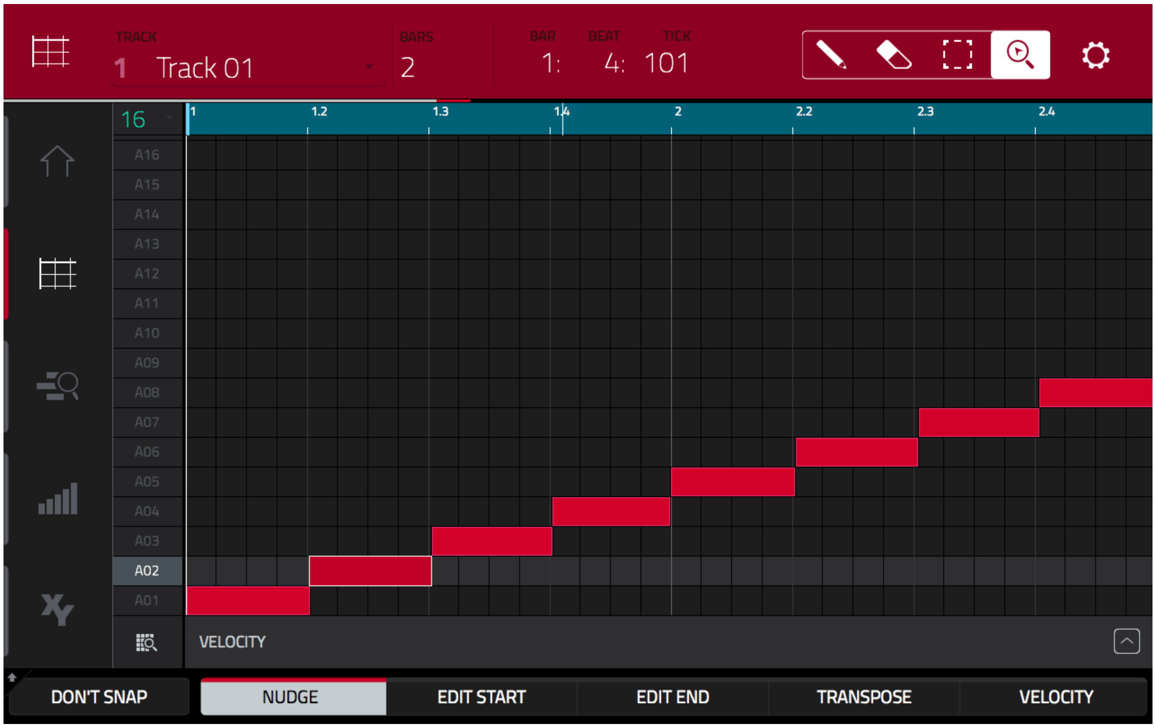
You can see that the MPC has created a new program, **Dirty_Breakch**, and also eight new samples, each representing one of the chop regions we set previously.

Close the PROJECT screen and take a look at **MAIN**:



The MPC has assigned the new **Dirty_Breakch** program to **track 1** of the current sequence (in fact it assigns it to the next available empty track in the current sequence). Hit pads **A01 to A08** to hear our individual break chops.

Now head over to **GRID EDIT**:



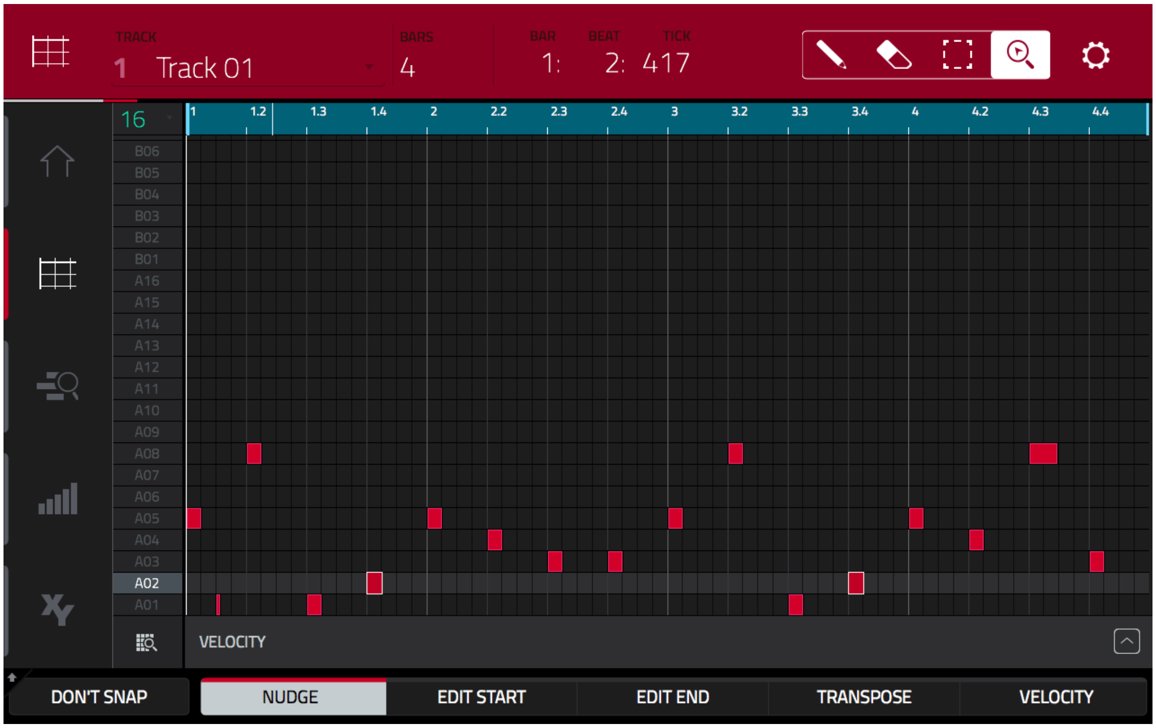
We have eight events, one for each chop region, so press **PLAY START** to listen to our break…. Yeah, sounds terrible. That's because the original break was 85 BPM while your default sequence is likely to be 120.0 BPM. Go back to **MAIN** and change the tempo to **85.0**. Now press **PLAY START** again – that's better!

So why would we do all this in the first place? Well chopping your breaks into smaller chunks lets us easily rearrange the original break by playing the chop regions in a different order.

From the **chapter 015 folder**, load up the **Region Chopping.xpj** project file.

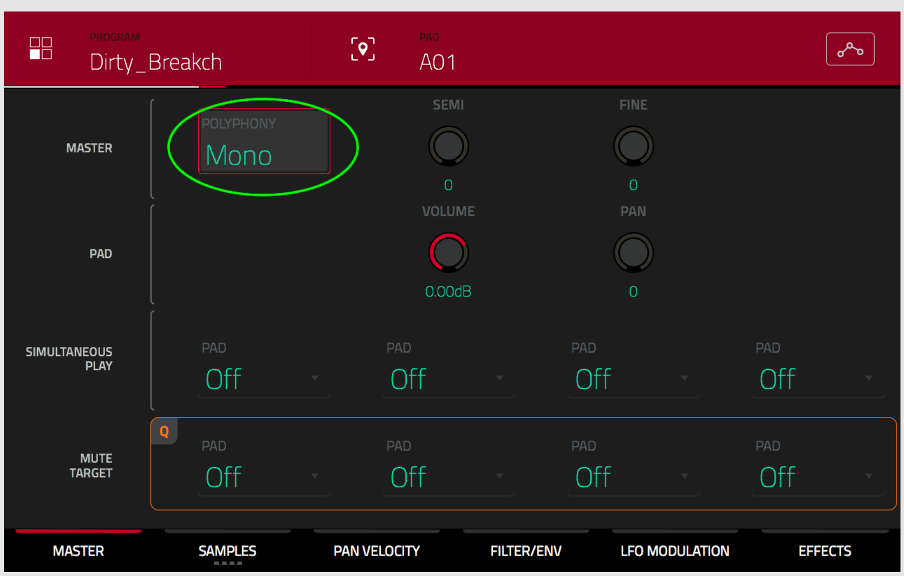
Sequence 1 is my version of the 'standard' sequence we just created. Select **sequence 2** and press **PLAY START** – this is my 'rearranged' version which I simply recorded in real time, finger drumming style, but you could record it using any method we've discussed previously.

As you can see from the grid, my new version has a very different event structure to the original.



Mono Program Playback

One thing you may have noticed when playing your chopped program is that it's not possible to overlap your chops – this is because a chopped program is set to a **POLYPHONY** of '**Mono**', as seen in **PROGRAM EDIT > MASTER**:



If you want to experiment with overlapping your chops (or overlapping any additional samples you might add to this program in the future) you should set the **POLYPHONY** to **Poly** (which is the default setting for a normal DRUM program).

016 Chopping Drums Part 2

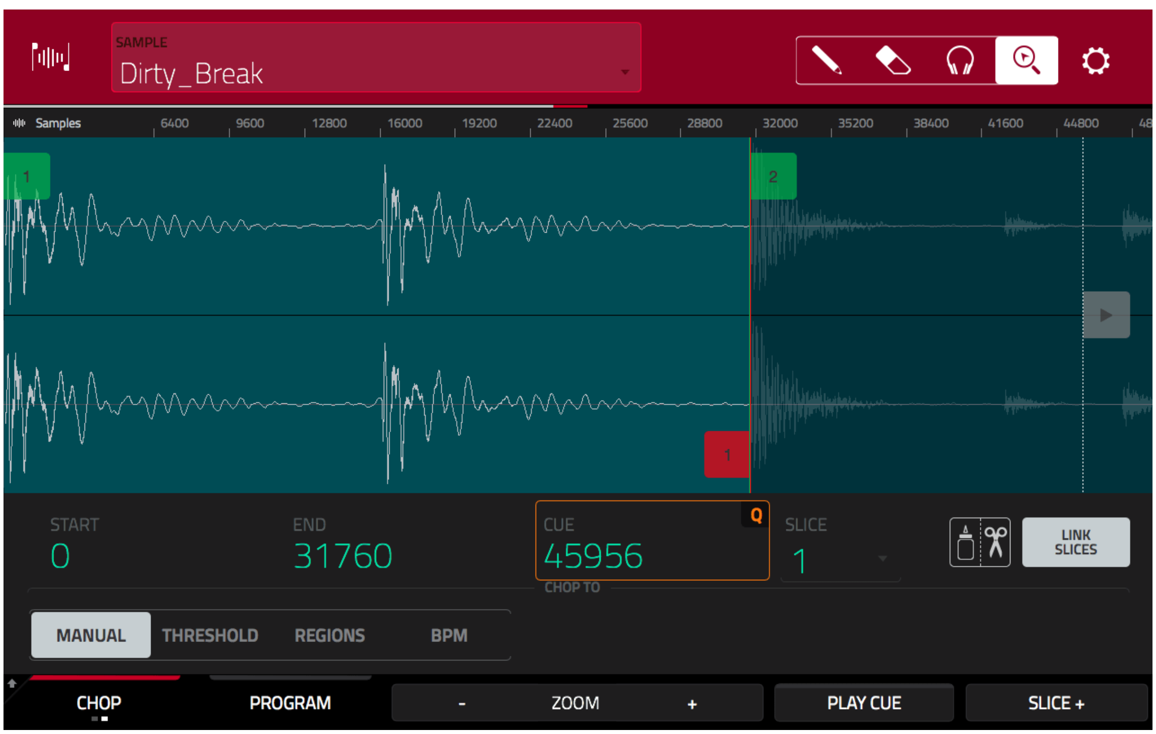
Chopping Down to Individual Hits

Region chopping like this is a great way to quickly 'remix' a break while retaining a lot of the original feel as each region still contains some 'timed' elements, leading to a natural sounding performance (e.g. region 1 contains two kicks, region 2 contains a snare followed by two hats and another snare, etc).

However as we heard when the sequence tempo was incorrectly set to 120.00 BPM, the fact that each region contains these small timed portions means we must always keep our sequence at the same tempo as the original break.

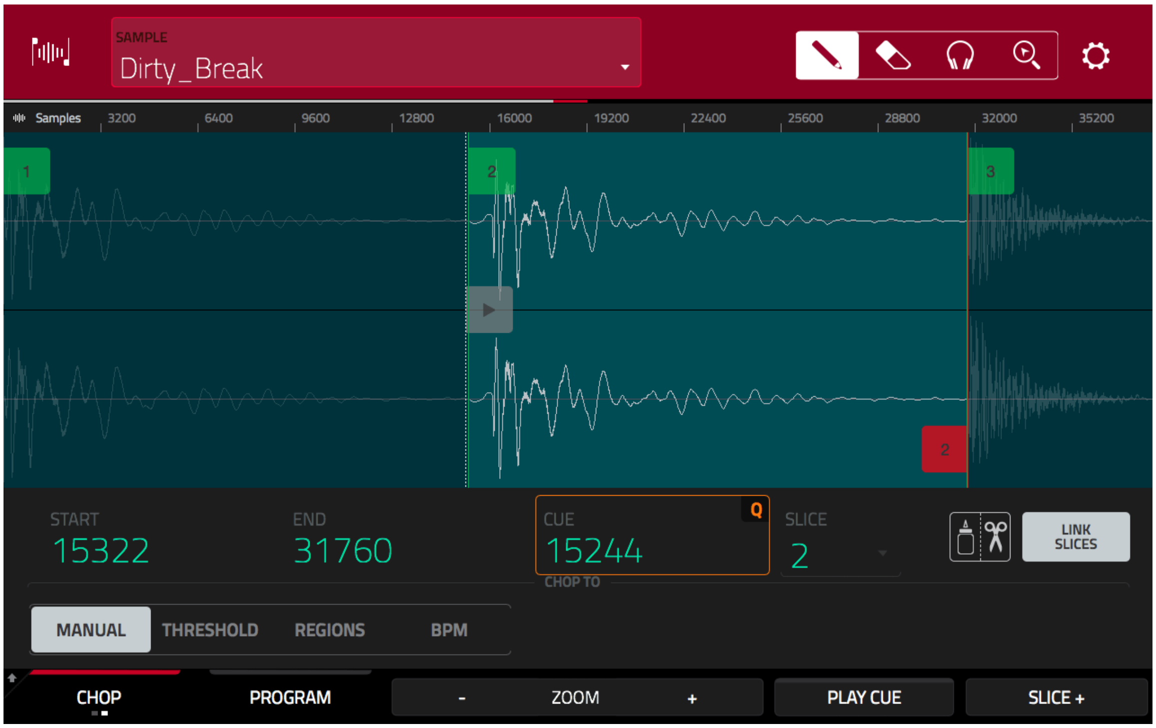
A more flexible method of chopping involves slicing our breaks all the way down to individual transients ('hits'). This way, each region only contains the one hit, there's no timed elements at all.

Head back to **SAMPLE EDIT** and select the **Dirty_Break** sample and zoom into **region 1**.



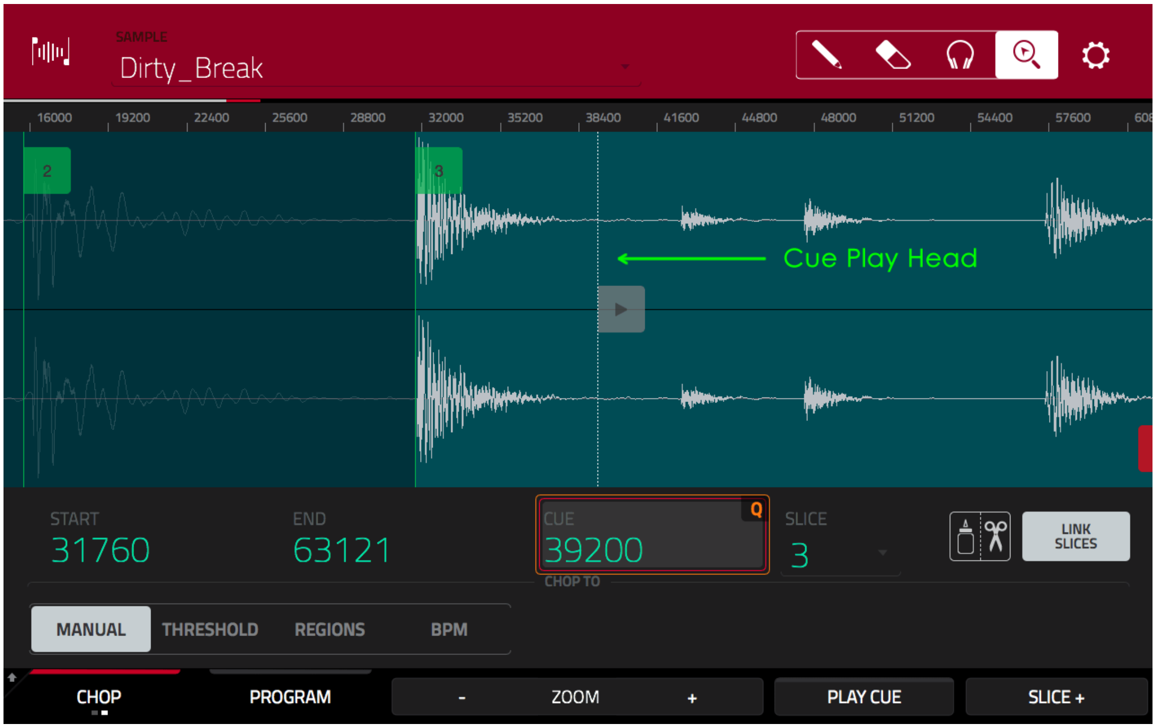
This region contains two kicks, so if we are chopping down to individual hits, we need to split this single region into two separate regions. A quick way to do this is to first select the pencil tool:

Now simply tap just before the second kick transient:

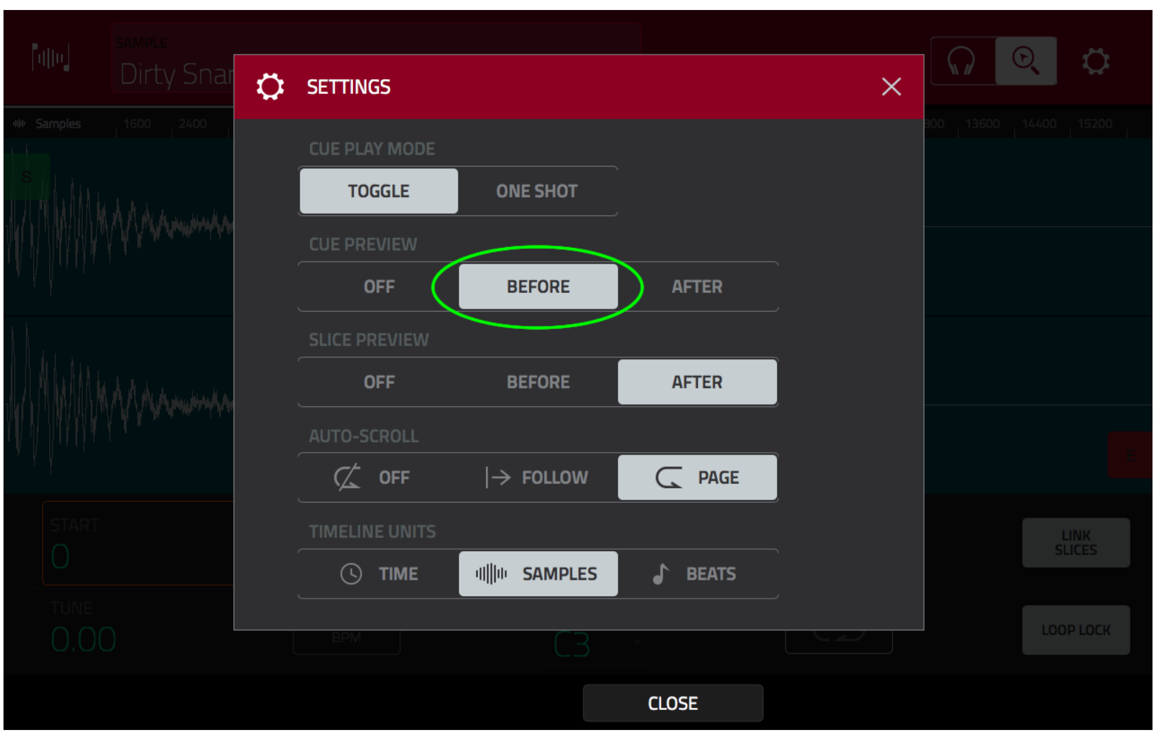


A new region is created at the point where you tapped; you can now use the usual editing options to adjust the START point of this new region so it is tight to that second kick.

Another option is to select the new **region 3** (hit **pad 3**) and start to turn the **top Q-LINK dial column 3 clockwise** (alternatively, tap the **CUE** parameter and turn the data wheel). Notice the new line on the screen?

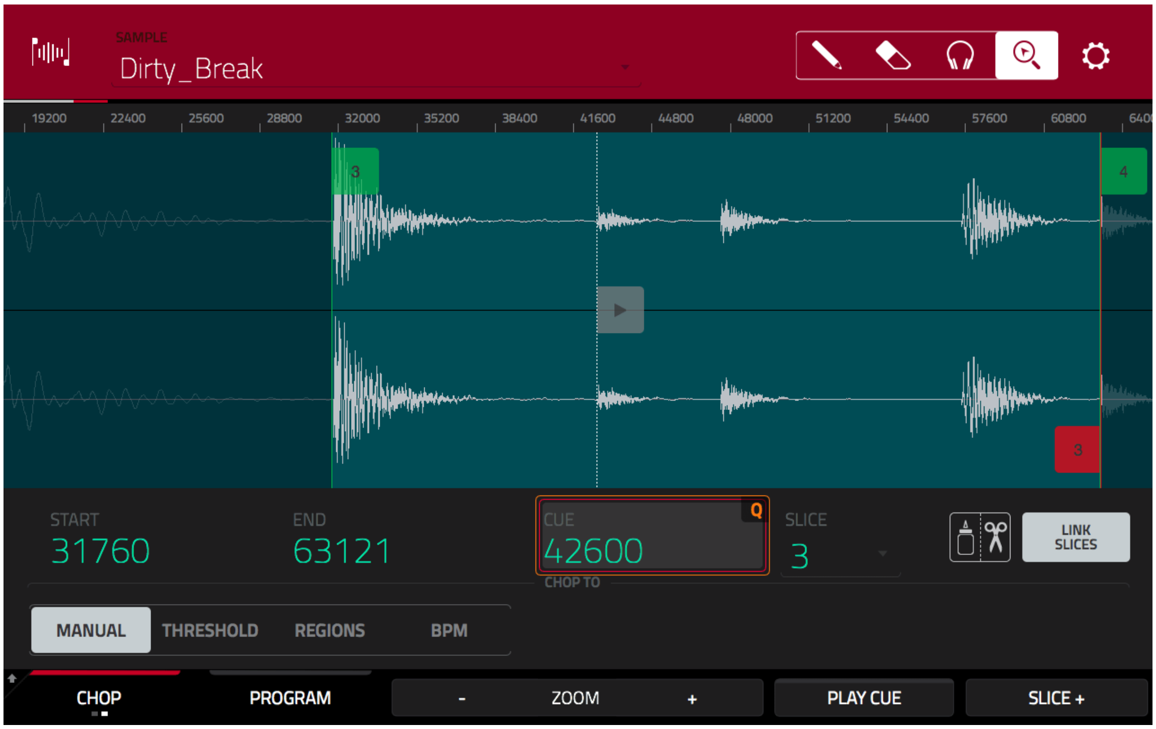


This is the CUE play head. Go to the **SETTINGS** 'gear' icon (top right of the screen):



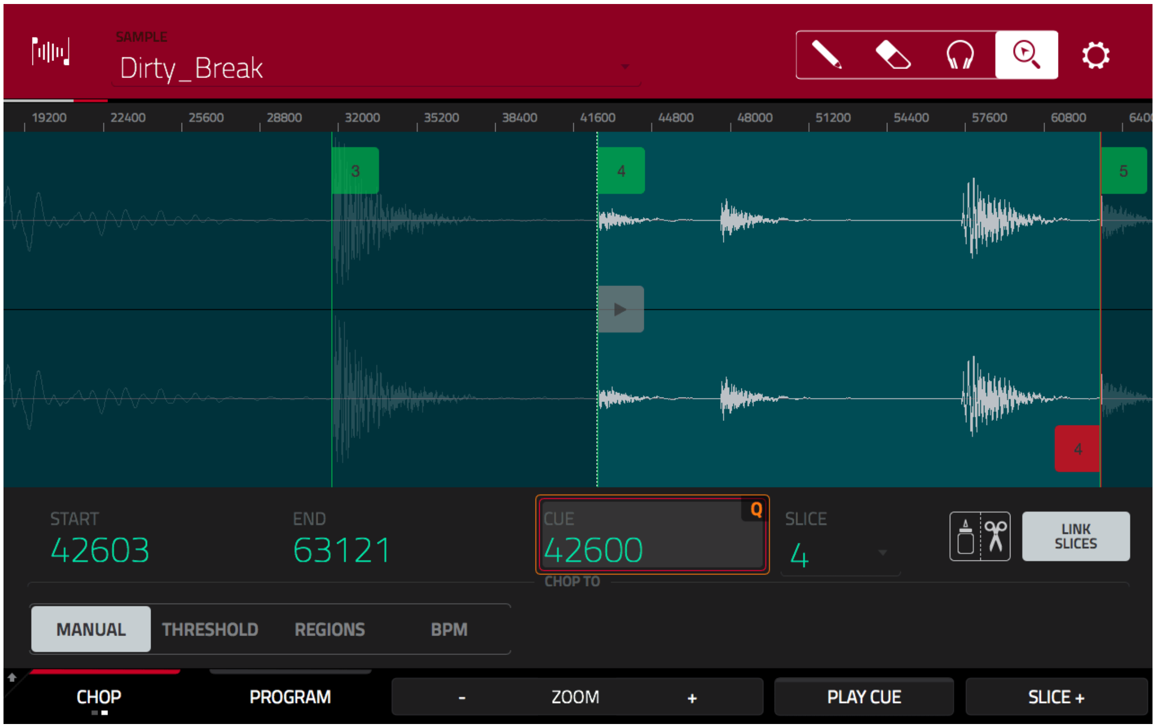
Set '**CUE PREVIEW**' to **BEFORE**. Above is the **CUE PLAY MODE** option – set this to **TOGGLE**. Press **CLOSE** and now scroll the cue play head through your waveform and you'll hear the audio at the CUE playhead position.

Continue moving the CUE play head until it is just before the first hi hat hit in this region (**42600**):



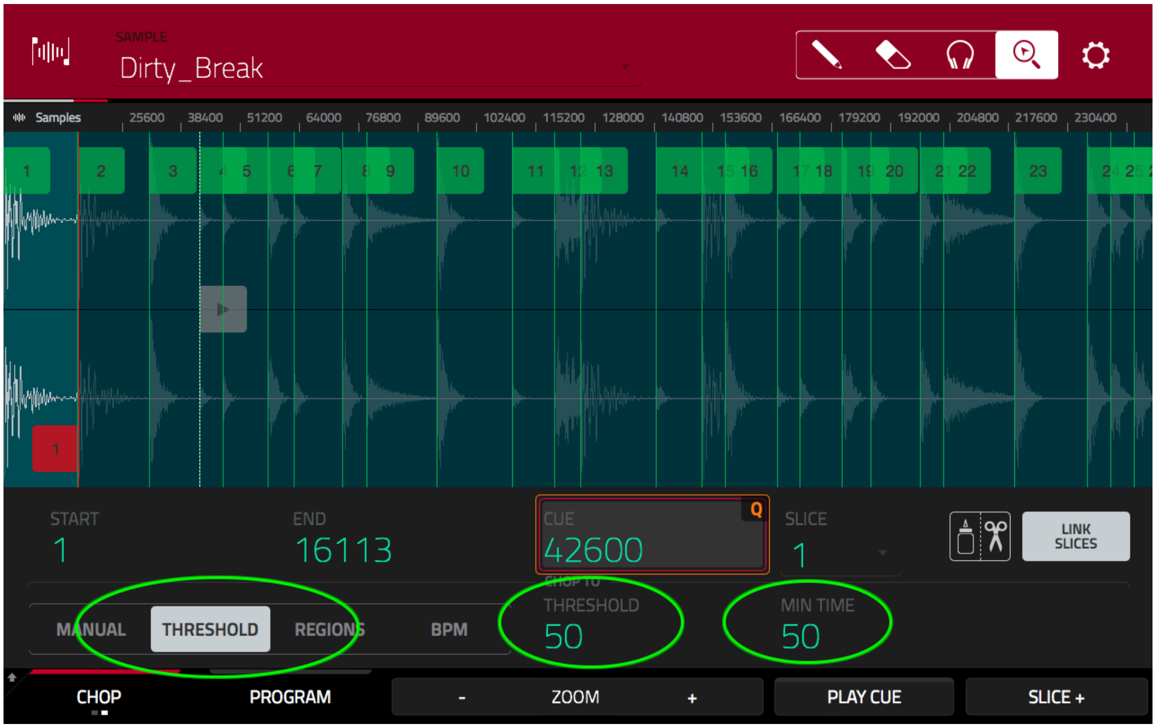
To hear what the audio from our cue point sounds like, press the **PLAY CUE** button at the bottom of the screen – to stop the waveform playing for too long, hit PLAY CUE again (this is where the TOGGLE option above comes in).

If you are happy with the CUE position, simply hit **SLICE+** and a new region is created at the CUE point.



This method offers more accuracy before adding the slice point. This way of manually adding chops is great for adding an extra region here and there, but not very practical when you are looking to chop dozens of regions in a break.

So, let’s make life even easier. Under the **CHOP TO** options, select **THRESHOLD**:



If your screen does not look the same as above, make sure the **THRESHOLD** and **MIN TIME** parameters are both set to **50.**

The MPC has now detected (nearly) all the individual drum hits in the break. There's now **26 slices** – to preview slices 17 – 26, select **BANK B**.

When in '**Threshold**' mode, the MPC Software is detecting peaks in the waveform above a specified level and assuming that these peaks represent unique drum hits within the break. This minimum audio level is referred to as the '**THRESHOLD**'. A setting of **50** works well for most drum breaks.

MIN TIME is the minimum amount of time the MPC should wait before looking for the next peak. A setting between 50-100 works well here for most breaks.

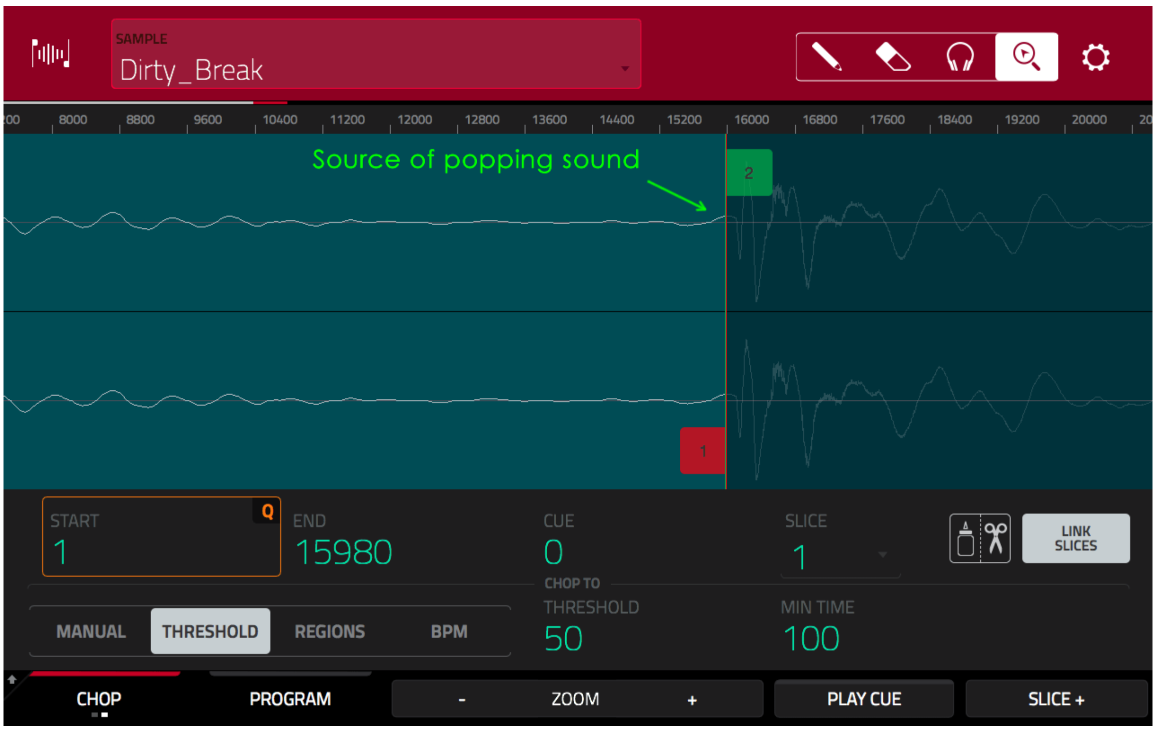
And if you do ever have problems with some hits being detected, you can always add additional slices manually.

Dealing with flams and drags

Region 12 in our break is a very short and quick drum roll, which is often referred to as a 'drag'. Even short versions are called 'flams'. While these are actually composed of multiple drum hits they are hit so quickly that in all practicality they are fine to be considered as a single hit, so I don't try to chop these down.

Fixing Clicks & Pops

Threshold chopping is great, however there is an annoying bug – some regions have an annoying 'click' or 'pop' at the end – for example, **pad A01**. Zoom in on the end of **region 1**:



You can actually see the source of the click at the end of the waveform. We can of course reduce the end point of region 1 so it stops before the click begins, however this will also affect the start point of region 2, which to my ears, actually sounds pretty tight, so I don't really want to change this.

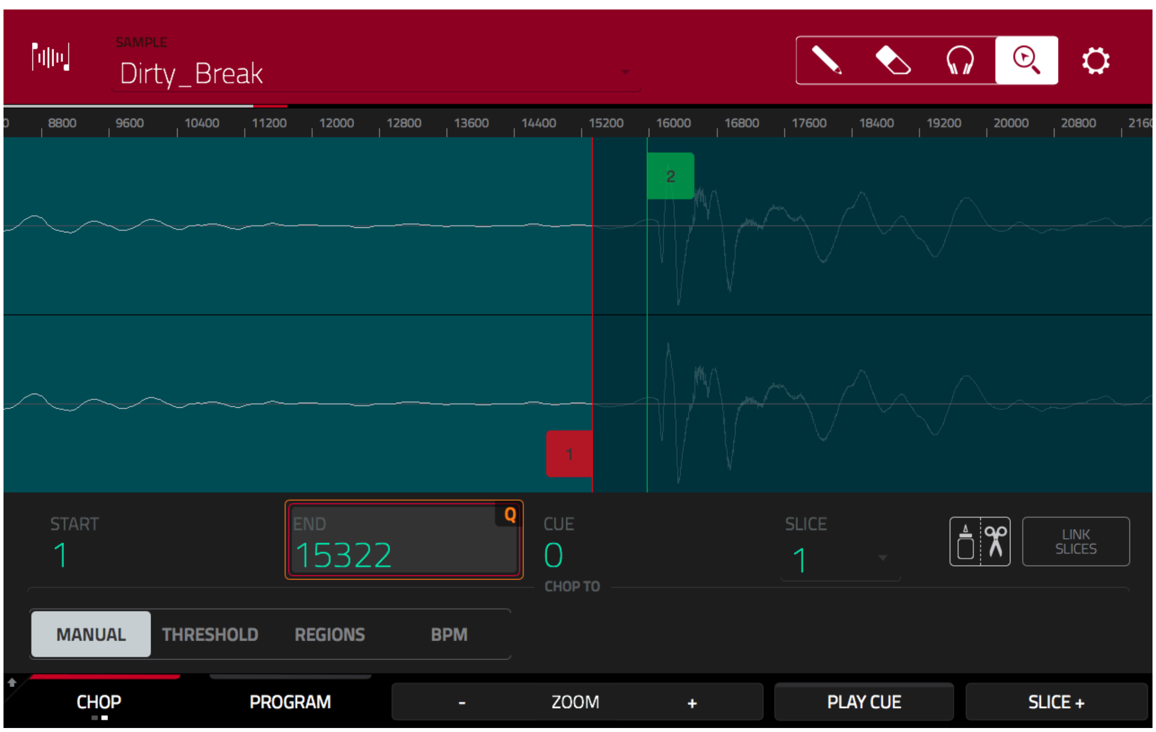
To get around this problem, we have to first disable **LINK SLICES**:

Tap **LINK SLICES** so it is no longer white.

With LINK SLICES disabled, we no longer have a 'shared' edit point between adjacent regions. Instead the START and END point of each region in our break is now entirely independent of any other regions.

Hold down **SHIFT** and turn off **SNAP 0** – this will allow us to make finer adjustments to our end point. With **region 1** selected, start turning the **second Q-LINK from the bottom**, **Q-LINK column 2** ***anticlockwise*** (this is the '**x10**' END point Q-LINK). Or if you prefer, use the **bottom Q-LINK, column 2** to move in '**x1**' increments.

As you turn the Q-LINK, keep on hitting pad 1 to continually preview the audio of region 1 with pad A01 – when you hear the click disappear, you're done. I set mine to **15322**.



As shown in the graphic above, the red END point of region 1 is now fully detached from the green START point of region 2, we've only edited region 1 leaving region 2 completely untouched.

You can now whizz through the rest of the regions and make similar adjustments to the end points of any regions with clicks at the end, for example, regions 3, 8, 13, 14, 16, 21 and 23. You'll soon find you can perform this kind of tweaking very quickly.

The key point is to use your ears – when the audible click is gone, you're done so quickly preview the next region, fix it (if necessary), move on. And so on.

Workflow Tip: Tweaking The Start Point

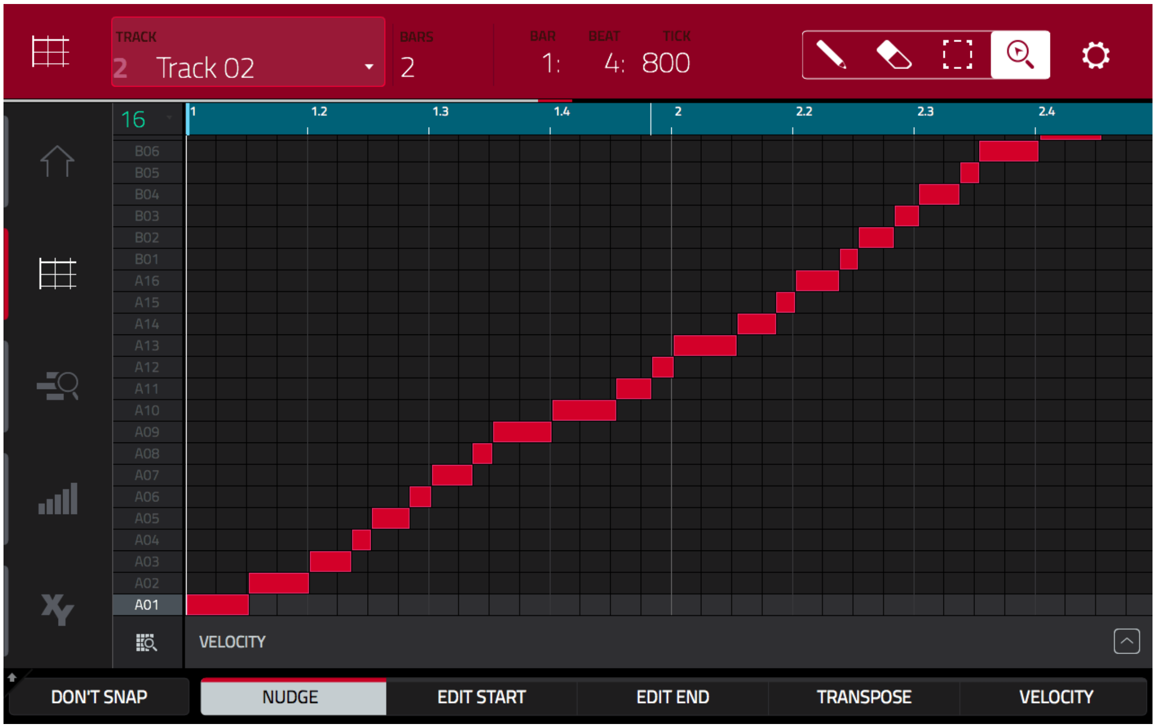
While you are editing the end points with the Q-Links, why not first make sure that the START parameter has been selected (single tap), this way if you think the START point also needs some adjustment you can use your data wheel.

This method ensures you keep your Q-LINK editing for the end point and data wheel for the START point, so there's no confusion, leading to a much quicker workflow (especially for Live users who would otherwise need to keep selecting different Q-LINK columns for start and end point editing).

Exporting Individual Hits

Go to **MAIN** and select the next blank (unused) sequence. Set a **TEMPO** of **85.00 BPM**. Now head back to our break in **SAMPLE EDIT**, hold down **SHIFT** and hit **CONVERT**, and as before, select '**New program with new samples'** and hit **DO IT** to create a new program with cropped samples and events.

Go to **MAIN** and press **PLAY START**. You should hear your chopped beat play back perfectly. As with region chopping, the MPC has created a program of chops (this time called **Dirty_Breakch-1**) and created sequencer events to recreate the break perfectly:



Load up the project file **Threshold Chopping.xpj** from the **chapter 016** folder. Select **sequence 1** (**Original Break**) to hear my version of this threshold chopped break.

Compared to the region chopping export we performed in the previous chapter, each event here only represents a *single drum hit* which naturally gives us a much more flexible sequence and program to work with. Preview all the slices by hitting your pads.

Real time Tempo Changes

In **MAIN** tap the **TEMPO** field and while your sequence plays, turn the data wheel to start increasing the sequence tempo. Unlike region chopping, this time our break matches the new tempo perfectly.

Try reducing the tempo so it is slower than the original 85.0 BPM. Again your break adapts nicely to the slower tempos, although eventually it becomes obvious that there are gaps between each chop.

This happens because ultimately all you are doing is firing off a bunch of slices from a break, and each one of those slices will contain background noise and hiss as well as the original room ambience it was recorded with. The moment that slice reaches its end point, all that hiss and noise also stops. If the next slice is triggered immediately you'll never notice this noise, but as you drop the tempo of your sequence the space between each event *increases* – this space will be true silence (there's no sampled audio there at all), so eventually it will become very obvious where the different slices are being triggered. This is why slowed down breaks can become 'choppy' sounding.

Configuring Your Sequence Before Export

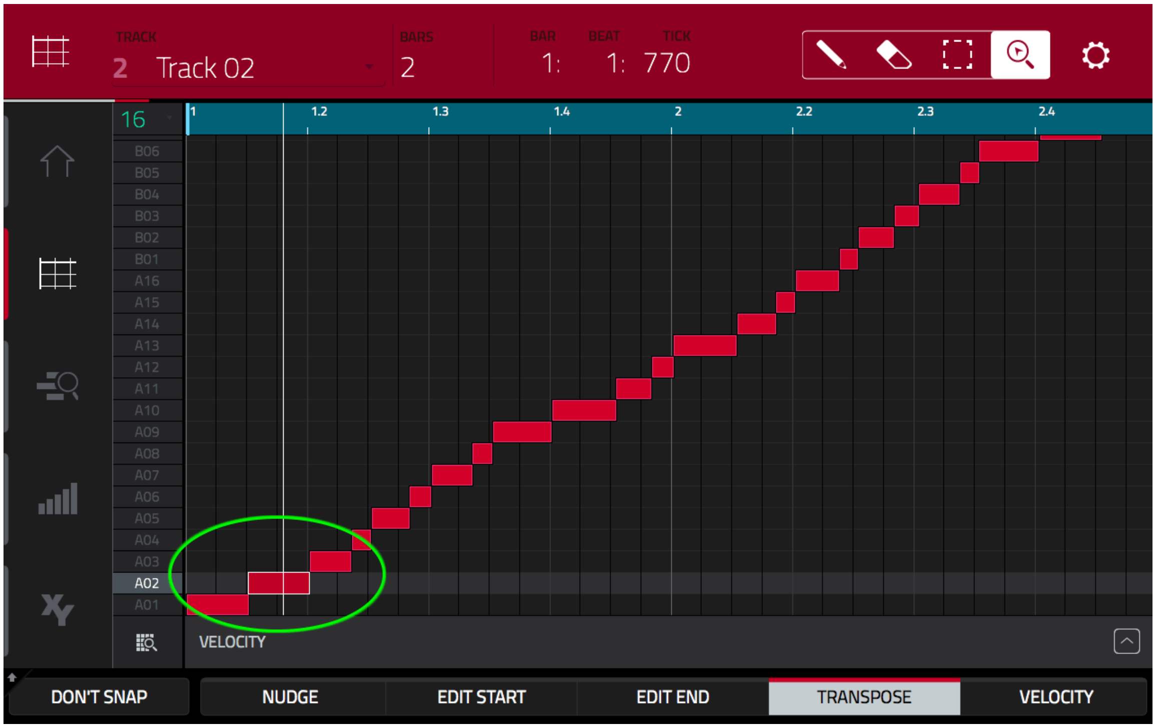
In the example I've shown here, we had a 2 bar break which we exported to the default blank MPC sequence in a new project – this also happens to be 2 bars in length. However it's important to note that when you are exporting a chopped break, there must be enough bars in the 'receiving' sequence to actually hold all these incoming MIDI events.

So, if you are exporting a 4 bar break, the currently selected sequence in your project must be at least 4 bars long. If it were, say, 2 bars long, only the first two bars of your chopped events would be exported to the sequence.

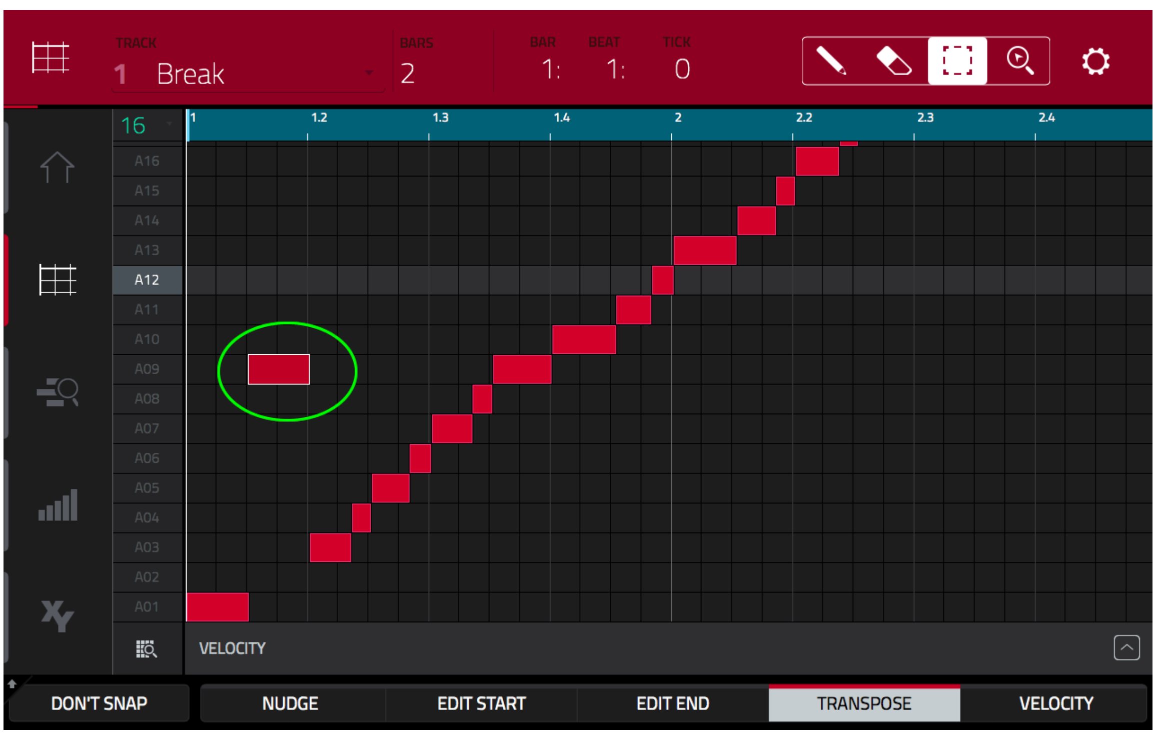
Select and play **sequence 2 ('Faster Tempo')** from the current project to hear my version of the faster tempo sequence.

Editing Existing Chop Events

Consider the second kick in this beat, which is on pad **A02**. It's this MIDI note here, on the pad **A02** line:



Want to change this to be an open hi hat instead? Pad A09 is an open hi hat slice, so let's just change this A02 event to be an A09 event instead. Hit pad A02 to select it, press the **TRANSPOSE** button and turn the data wheel **seven clicks clockwise** (or press the + button seven times)



Press **PLAY START** to hear the difference – how easy was that? You can hear my version on **sequence 3** of the **Threshold Chopping** project, where I've transposed several other events across the sequence, including a straight swap of the high pitched snare (pad A3) with a lower pitched version from A16.

Remember this is just a standard sequence so you can perform nudges and velocity changes to any event in your GRID.

Overdubbing Additional Chops

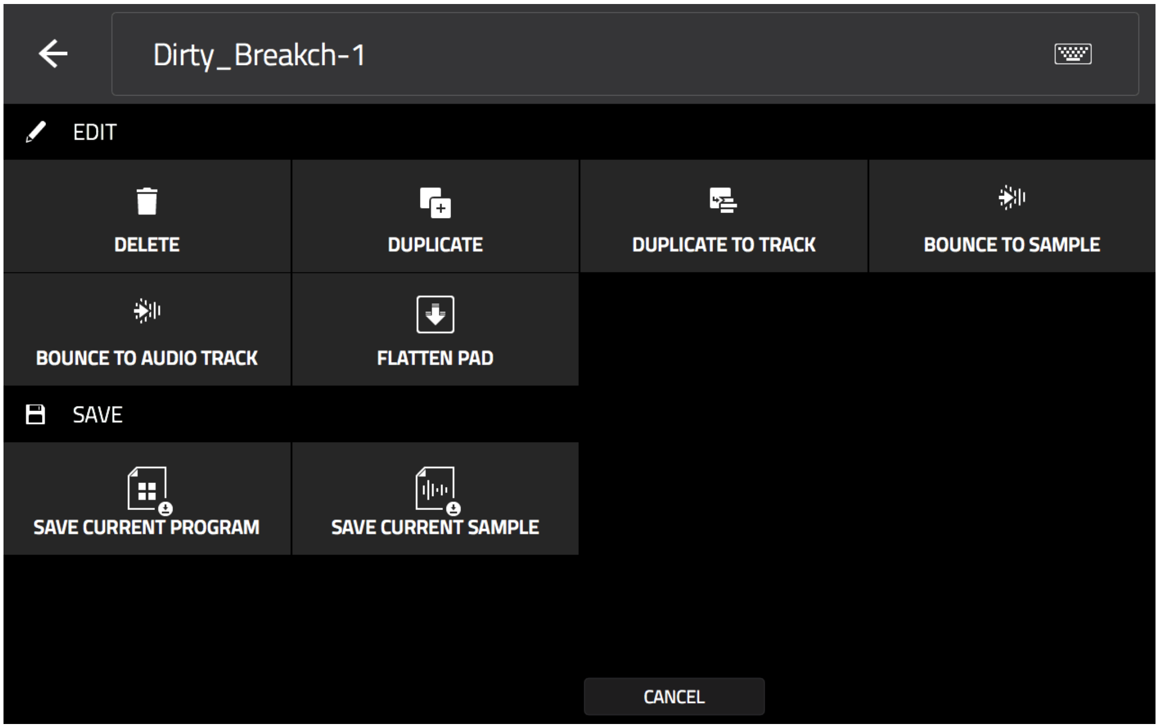
Let's say you wanted to add some additional kicks to your beat to add a bit more groove and feel these can be added in real time with quantise turned off. For example we can use the kick found on pad **A02**. To turn off quantize, press **TC** (top right of MAIN screen) and set **TIMING CORRECT** to **OFF**.

Remember I mentioned before that chopped programs are automatically set to MONO (monophonic) playback? Now's the time where we need to set this back to POLY (polyphonic) playback as this will allow us to overdub our ghost hits over other events. Go to **PROGRAM EDIT > MASTER** and set **POLYPHONY** to **Poly**.

Before recording any overdubs to an existing sequence, I would normally have a bit of a practice run. Just press PLAY START and while the beat is playing try experimenting with adding some groovy ghost kicks to your beat using pad A15.

Once you are happy, hit **OVERDUB** and **PLAY START** and record your ghost kicks to your sequence. If you make a mistake, hit **UNDO** and re-overdub it (or manually edit the incorrect note events in the Grid Editor using the NUDGE tool). Remember you can turn off **FULL LEVEL** to help vary the velocity of your ghost notes, or edit the velocity of recorded notes in the Velocity lane. Repeat the process with some additional snare hits.

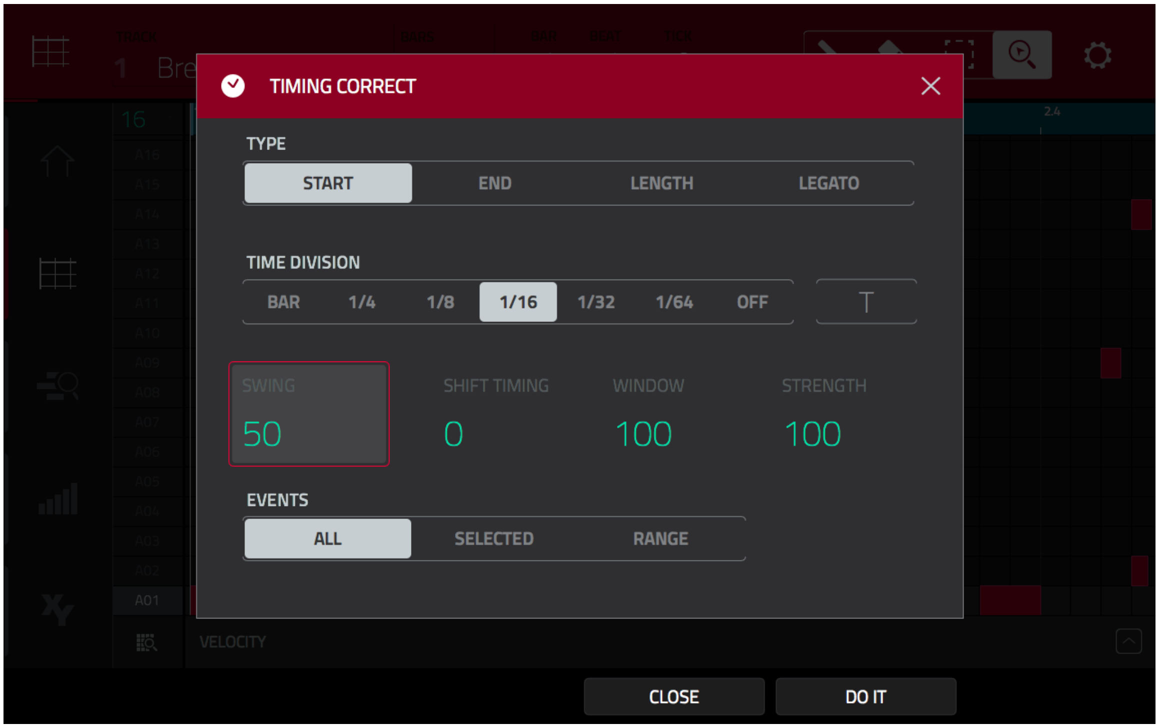
Select **sequence 4** (**Overdub**) to hear my version – notice how I have made a copy of the original chopped break program (**Dirty_Breakch-2**) in which I have set all pad playback to POLY. You can easily duplicate a program in MAIN by hitting the **pencil icon** at the end of the **DRUM PROGRAM** row and selecting **DUPLICATE** (or **DUPLICATE TO TRACK** to simultaneously copy and assign the new program to the next available track).



Applying a Quantize Template

While you were recording your overdubs, you may have noticed that the break wasn't perfectly matching your metronome clicks. The reason for this is simple – your original break was recorded by a human, not a sequencer, and as such the original drummer would have introduced his own unique feel to the performance. A live drum performance will rarely conform to a rigid 'quantise' template – however with the performance now converted into MIDI event data, we can use the timing correct features to apply a quantise template, just like we looked at in the Quantising & Swing tutorial.

In **GRID VIEW**, hold down **SHIFT** and press **T.C.**:



Set the **Tim Div** to **1/16** (i.e. a quantize template of 1/16) and set '**Events'** to **All**. Press **DO IT** and the MPC will shift all your events to the nearest 1/16th quantise point. A setting of **1/16** is a very good setting for a nice tight sequenced feel. Follow the green arrows below to see how some of these notes were moved.

Press **PLAY START** to hear your beat now. It's lost that 'human' feel, and is definitely sounding more 'sequenced' with a tighter feel.

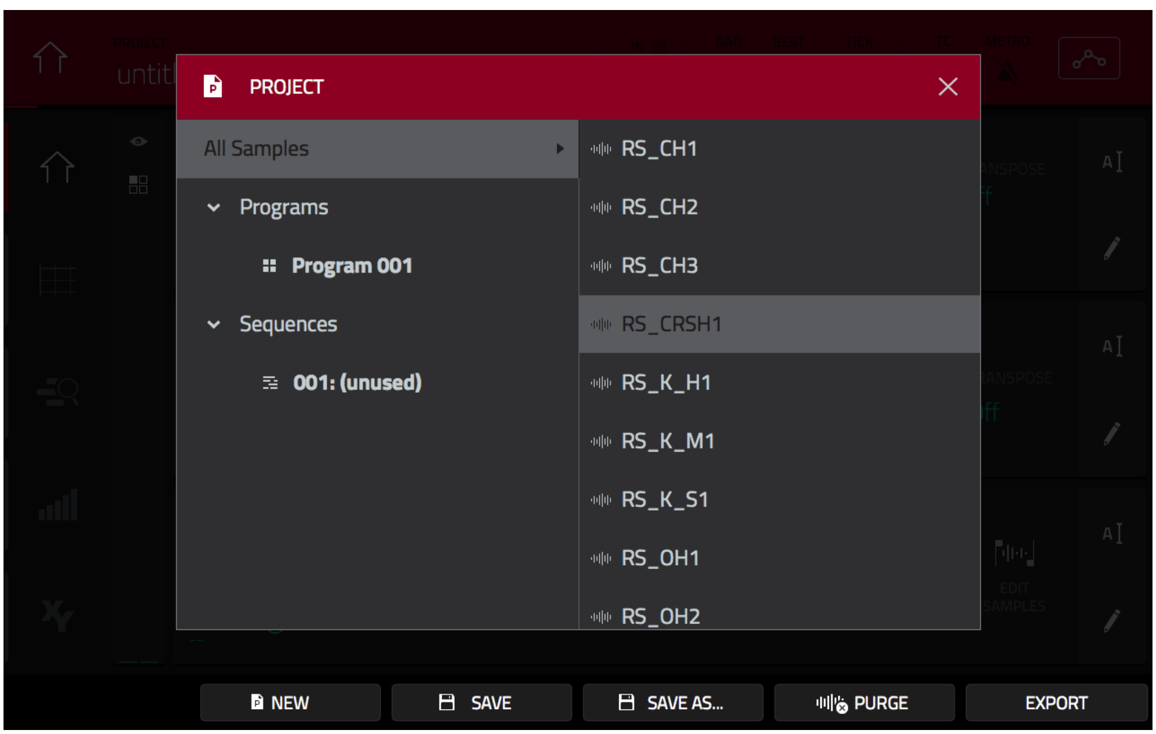
You can hear my version in **sequence 5** (**Quantize**). Of course applying a quantize template has also quantized my ghost notes, so if you wanted the bulk of your beat to be quantized, but also have some un-quantized ghost notes, you would be better off applying a quantize template first and then recording your ghost notes after (with Timing Correct set to Off).

017 Building Velocity Sensitive Kits & Sequences

So far we've been using the auto-generated 'chop' program as the basis of our chopped sequences. In this tutorial we'll build a very realistic sounding multisampled 'live' drum kit using samples 'threshold chopped' from live drum breaks.

First, start with a new blank project (**MAIN > PROJECT folder icon > NEW**). In the **BROWSER,** navigate to the **chapter 017** folder and single tap the '**Multisampled Drums**' sub folder. Inside are the drum hits 'threshold chopped' from a couple of live breaks (taken from the same studio session as the drum break I've been using in previous chopping tutorials).

Hold down **SHIFT** and press the **LOAD ALL** button to load the entire contents of this folder into your project. Go to **MAIN** and tap the current project name to open the **PROJECT** panel:



With **ALL SAMPLES** selected you can see our multisampled drums and there's a few examples of each instrument. All kicks and snares are provided with multiple 'takes' at three different velocities'; soft (S), medium (M) and hard (H).

- **RS** is simply a unique prefix for these sounds
- **K** stands for kick, **SN** stands for snare, **CH** is a closed hat, **OH** is an open hat.
- **H** stands for 'hard', as in 'hard hit'. **M** is a medium hit, **S** is a soft hit.
- **3** means this is hard kick number 3 in the collection.

The closed and open hit hats are all recorded at the same approximate velocity, but all sound a little different. These samples are just in the format **RS_CH_1**.

Close the PROJECT panel and in the **MAIN > PROGRAM** field, locate the default blank **Program 001** DRUM program and rename it '**Finger Kit'** using the '**A**' icon at the right end of the row.

Timbre Variation Using Multiple Drum Sounds

One of the very cool features of DRUM programs is the ability to set up a drum kit that can be used to create realistic sounding live drum performances. One key element to creating something so realistic is that we can emulate the natural timbre variations you'd expect to hear in a real drum performance.

Timbre simply refers to the different ways an instrument can sound depending on how and where it is hit (or played). Take hitting a snare drum for example – think how differently that snare drum would sound if you first hit it very hard and then hit it very soft. There would not only be a difference in the volume, there would be a massive difference in the character of the sound.

A hard snare hit has a very pronounced 'crack' and a longer decay 'tail' (it will ring for much longer after being hit). A softly hit snare on the other hand will not really have any pronounced 'crack' and will not ring out for very long at all – it will also often sound duller. These changes in characteristics are referred to as a *change in timbre*.

You'll hear other obvious timbre changes when you strike a snare in different positions, such as in the middle compared to hitting it nearer the edge. However even if you try hitting in the same place at the same velocity ten times straight, it would most likely sound slightly different each time. It's these subtle (and not so subtle) changes in timbre that give life and realism to any drum performance you record in your MPC.

Using Multisampled Drums

An MPC DRUM program provides you with all the features you need to create realistic timbre variations, be it artificially emulating them or by using pre-recorded 'multisamples' of the same drum sound.

Firstly, let's assign a kick, snare, closed hat and open hat to pads A05 – A08. These will be our primary sounds on layer 1 of each pad. So, go to **PROGRAM EDIT > SAMPLES** and assign **RS_K_H1** to **A05**, **RS_SN_H1** to **A06**, **RS_CH1** to **A07** and **RS_OH1** to pad **A08** using the techniques discussed previously in the book.

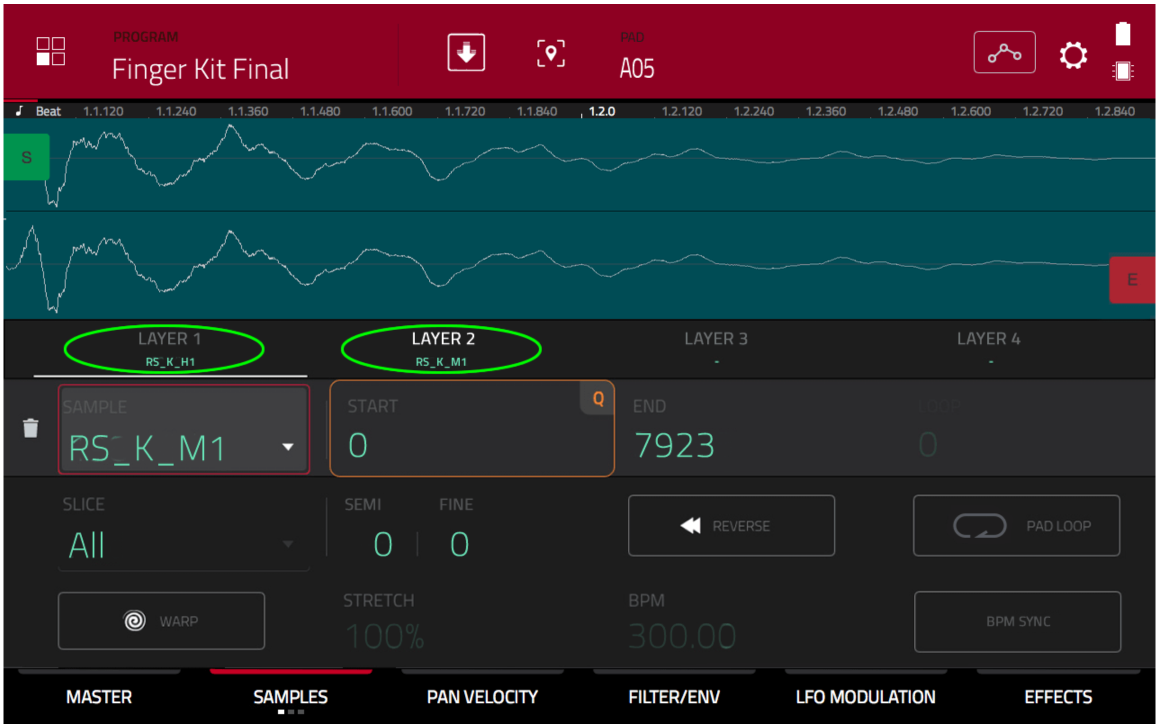
Now, in our newly loaded folder of sounds we also have some medium and soft hit kicks, so wouldn't it be great if we could have the MPC play the hard kick for 'hard' pad hits, the medium kick for 'medium' velocity pad hits and the 'soft' kick for soft pad hits? Welcome to the world of **velocity switching** and **dynamic layers.**

Velocity Switching Pad Layers

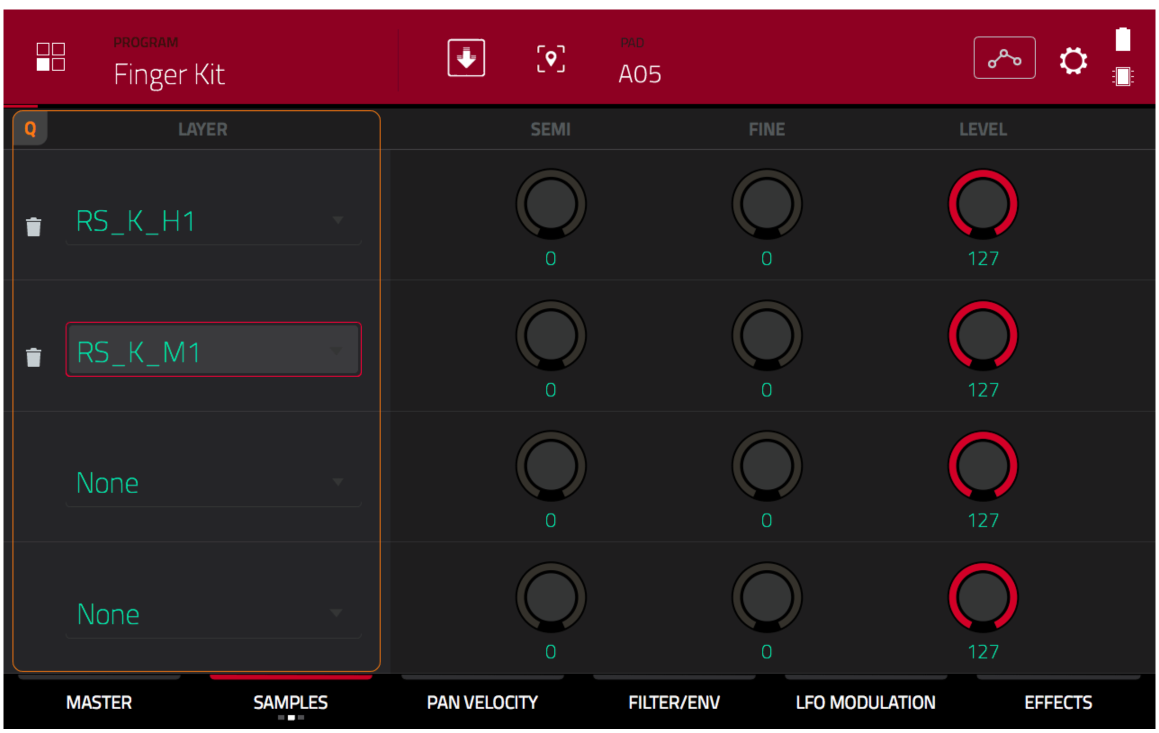
So far we've assigned a single sample to each pad, but we know from the Drum Layering chapter, each pad in an MPC DRUM program can hold up to *four* samples – one assigned to each pad 'layer'.

Now by default, when you assign more than one sample to a pad it will play back all the samples simultaneously, which was very handy for drum layering. However we can also configure a pad to play back just *one* of the layers based on certain criteria. One of these criteria is based on the *velocity* at which the pad is hit. So, if we hit the pad softly, we can tell the MPC to play back a soft sample on layer 3, if we hit it with medium velocity it plays the 'medium' sample on layer 2, and if we play it hard, it plays back the 'hard' sample on layer 1. This is called **velocity switching**.

Press pad **A05**. The **RS_KIK_H_1** sample is already assigned to layer 1 of this pad – this is the 'hard' kick. We now need to assign a 'medium' kick to layer 2 of this same pad – let's use the sample **RS_K_M1** ('M' for 'medium'). If you want to use the Q-links for this, using the second Q-Link from the top in Q-Link bank 1. However I prefer to just tap on the second layer and turn the data wheel until you find the **RS_K_M1** sound from the sample list. You can perform this is either **SAMPLES screen '1'** :



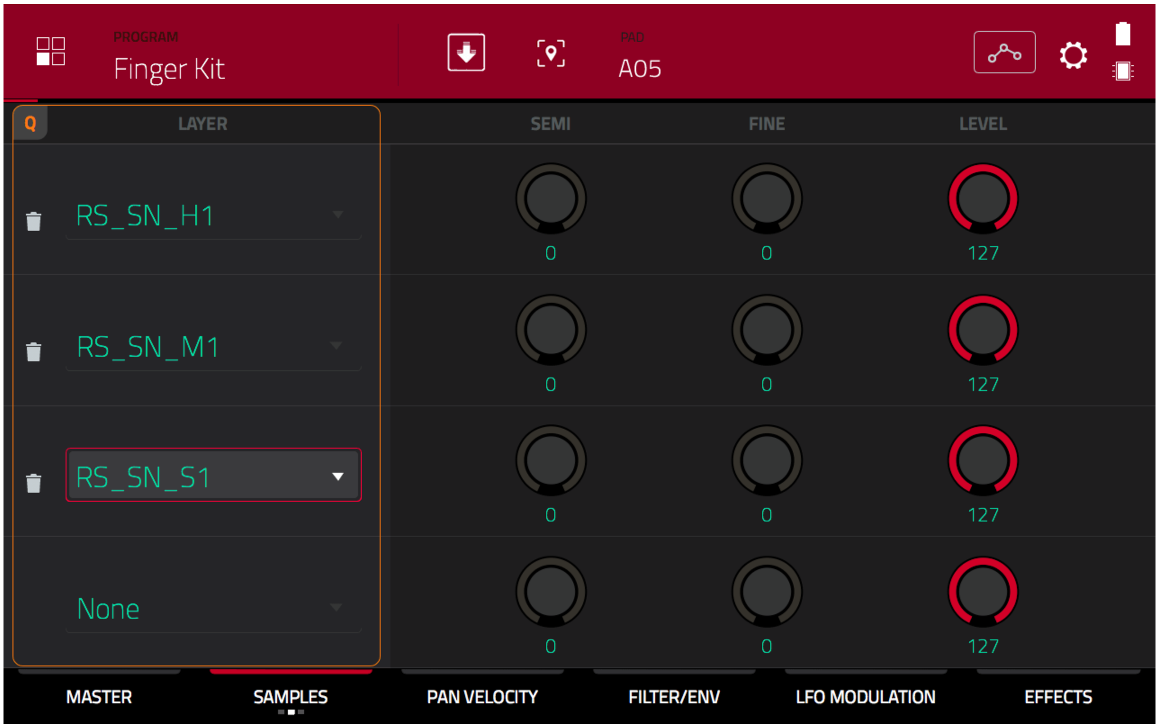
or **SAMPLES screen '2'**:



Repeat this procedure for **layer 3** – this time assigning the 'soft' kick sample **RS_K_S_1**

If you listen to pad **A05** now you'll hear all three samples playing *simultaneously*, and hence acting like a layered drum sound, but we'll fix that very soon.

Now select pad **A06** and follow the procedure we used for the kicks, but this time assign our three snare dynamics; so **layer 1** is **RS_SN_H1**, **layer 2** is **RS_SN_M1** and **layer 3** is **RS_SN_S1**.



Again, if you preview pad **A06** you'll just hear all three snares play simultaneously.

Setting Up the Velocity Switch Ranges

Now, to stop our kick and snare pads simply playing all three layers simultaneously, we need to set up 'dynamic switching'. Dynamic switching works by the MPC detecting how hard you hit the pad (i.e. the 'velocity') - based on your parameter settings, it will play back only ***one*** of the three layers you set up.

To tell the MPC which layer to play, you need set up **velocity 'ranges'** - to do this, go to the **PAN VELOCITY** screen for pad **A05**.

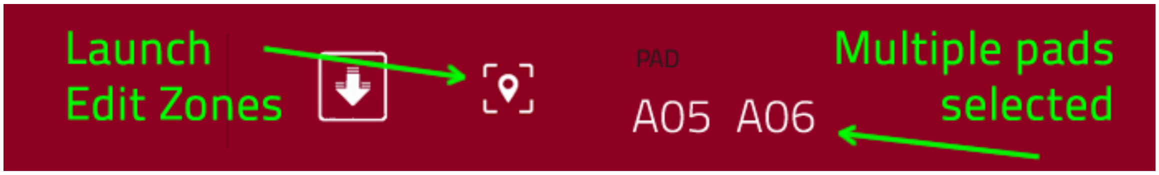
The velocity ranges are set in the **VEL START** and **VEL END** columns (Q Link columns 2 and 3 respectively) with VEL START handling the lowest velocity value, and VEL END the highest velocity value for this layer.

The idea here is to set a suitable range for each layer that doesn't overlap the range set for any other layer. So first we need to define the range for a 'hard' hit – for example, we could say a **hard hit** covers a velocity of **110 – 127** (remember the hardest hit on an MPC is represented by a velocity of **127**). A **medium hit** could cover a velocity range of **60 – 109**. And a **soft hit** would therefore cover everything below this – i.e. a velocity range of **0 to 59**.

These are just suggestions, there's no actual hard rule and you can set your velocity ranges to any values you wish based on your unique playing style; you'll also find your preferred velocity ranges will vary for each kit you build.

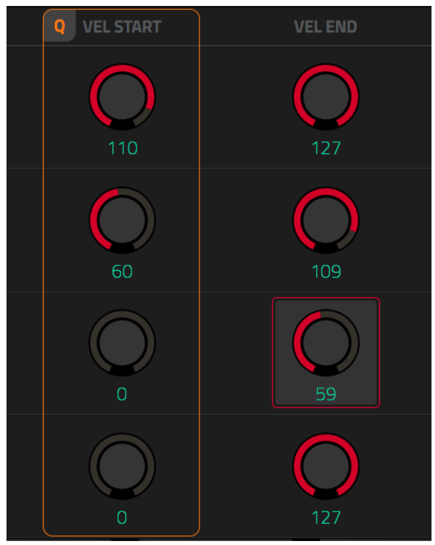
So now we simply need to enter these ranges onto the layers on pads **A05** and **A06**. First, as we are editing two pads identically, lets select them both so our edits can be applied to both pads simultaneously, saving time.

To do this hold down the **SHIFT** key, press pad **A05**, and press pad **A06.** Both pads are now illuminated and you'll see the **EDIT ZONES** icon change:



Now any changes you make to one of these pads will be mirrored in the other pad. Press **CLOSE**.

You can now set your velocity switch ranges; I find the Q-Links a bit over-sensitive in this screen so I prefer to single tap each value and use the data wheel. Set up your values as follows:



Leave 'MULTIPLE' mode by pressing the **EDIT ZONES** icon selecting **CURRENT**.

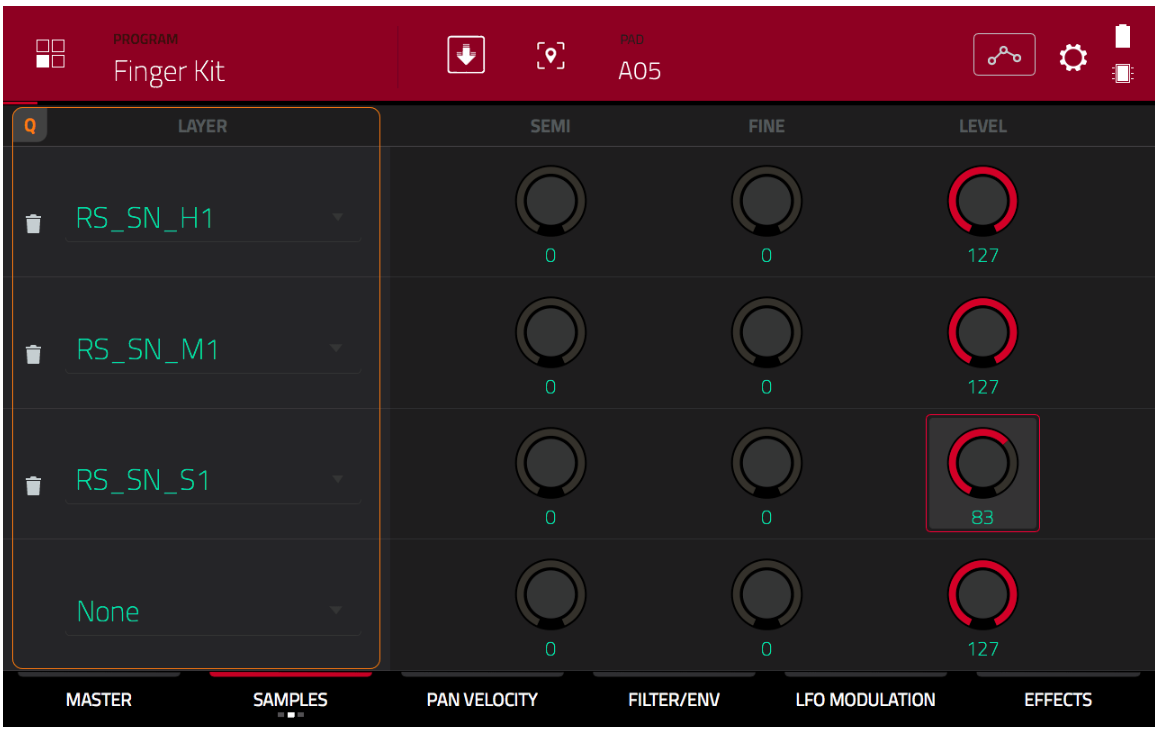
Press **A05** and then **A06** and observe in the **PAN VELOCITY** screen how both pads have identical velocity ranges set up.

Performing Velocity Switches

With your velocity switches configured, it's time to try them out. The most important step here is to turn **FULL LEVEL** *off*, otherwise all your pad hits will be at a velocity of 127 regardless of the actual velocity you actually strike the pad with! So make sure the FULL LEVEL button is not illuminated on your hardware, nor should it be flashing (HALF LEVEL, which is set by holding down SHIFT when pressing FULL LEVEL).

With FULL LEVEL off, start hitting pad **A05** at different levels of hardness from soft all the way up to hard - you should hear the different dynamic sample layers being played. At this point you may need to tweak the velocity ranges to suit your style of playing – for example if you tend to hit the pads fairly softly, then your ‘soft’ range may need to be set lower.

Another thing to consider is the ‘**Level**’ of each layer. It is important that your switching is smooth, and often if your sounds have been individually normalised, a soft snare can actually sound far too loud compared to one of the harder hits. To adjust any layers that appear too loud, simply change the ’Level’ setting for that specific layer. For example, I adjusted the ‘**Level**’ for the snare on pad **A06 level 3** to the following (**83**) – this is the last column in the **SAMPLES** screen:



For me, this gave a much smoother transition between layers.

Layer Switching – Creating ‘Cycle Kits’

In addition to the velocity switching option, we are also able to utilise the different layers on our pads by having the MPC play a different layer each time the pad is hit, *regardless* of the actual velocity you hit the pad.

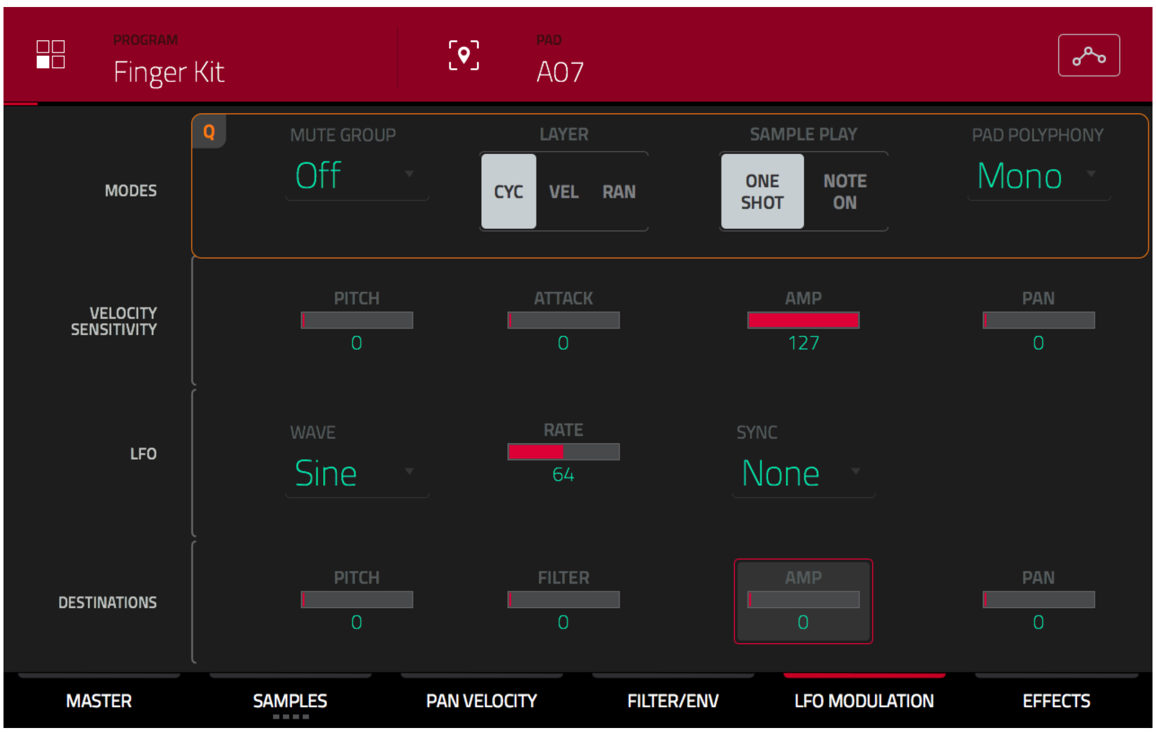
This method is perfect when you have a bunch of multisamples that were originally played at a similar velocity but with differing timbres. This option is independent of velocity, so even works with FULL LEVEL (and HALF LEVEL) switched on.

We looked at CYC and RAN earlier in the book when we set up a round robin for our layered vinyl crackle.

This time let's set up the closed hi hat on pad **A07**. All the closed hi hats we have are medium velocity hats, but each one sounds slightly different. Assign **RS_CH1** to **layer 1**, **RS_CH2** to **layer 2** and **RS_CH3** to layer **3**.

As before, these layers are currently set to all play simultaneously when the pad is struck. To change this behaviour, go to **PROGRAM EDIT > LFO MODULATION** and adjust the '**LAYER**' setting to either **CYC** (layer 'cycle') or **RAN** (random layer playback).

Either of these settings will work fine – you could argue that cycle will always ensure that each pad hit will produce a different sample every time, while with random there's a possibility that you could get the same sample play a few times in succession. But it's up to you – set **LAYER** to either **RAN** or **CYC**.



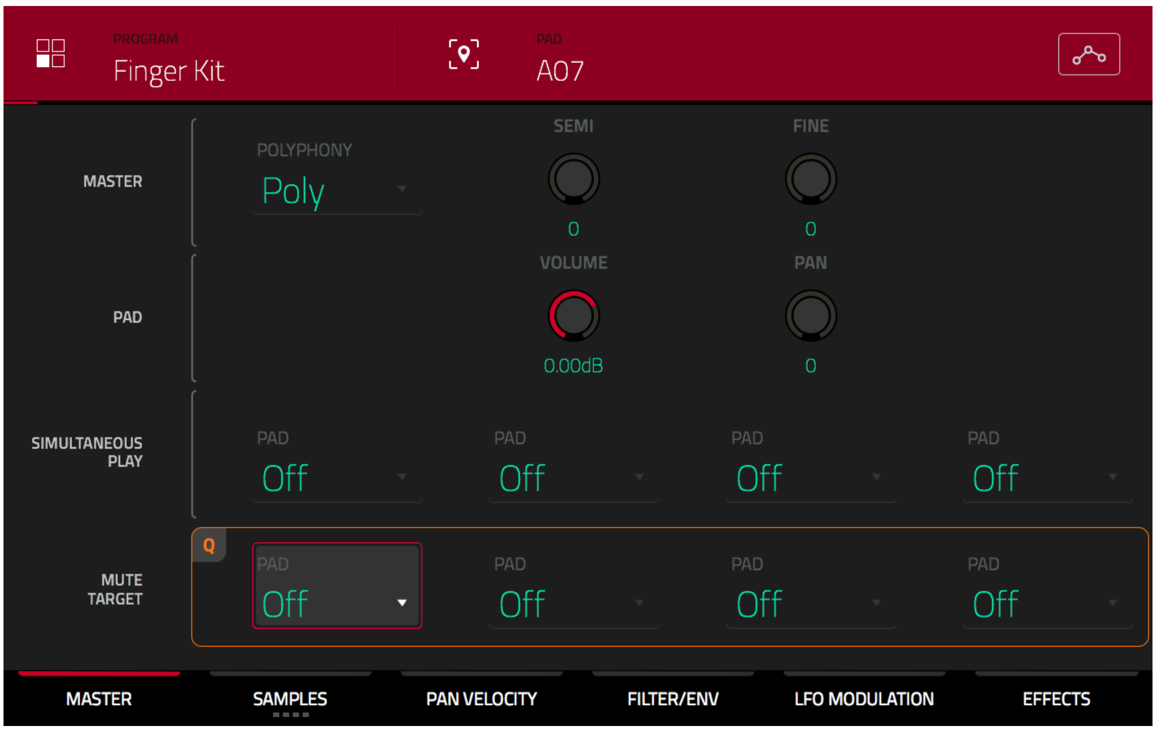
Now hit pad **A07** several times in succession to hear your cycling or random layer play in action. Sounds pretty good!

Remember if you had four samples you could use all four assigned to the four layers, and if you have two, you can just use those two – it doesn't matter if some layers are blank, the MPC will just ignore them.

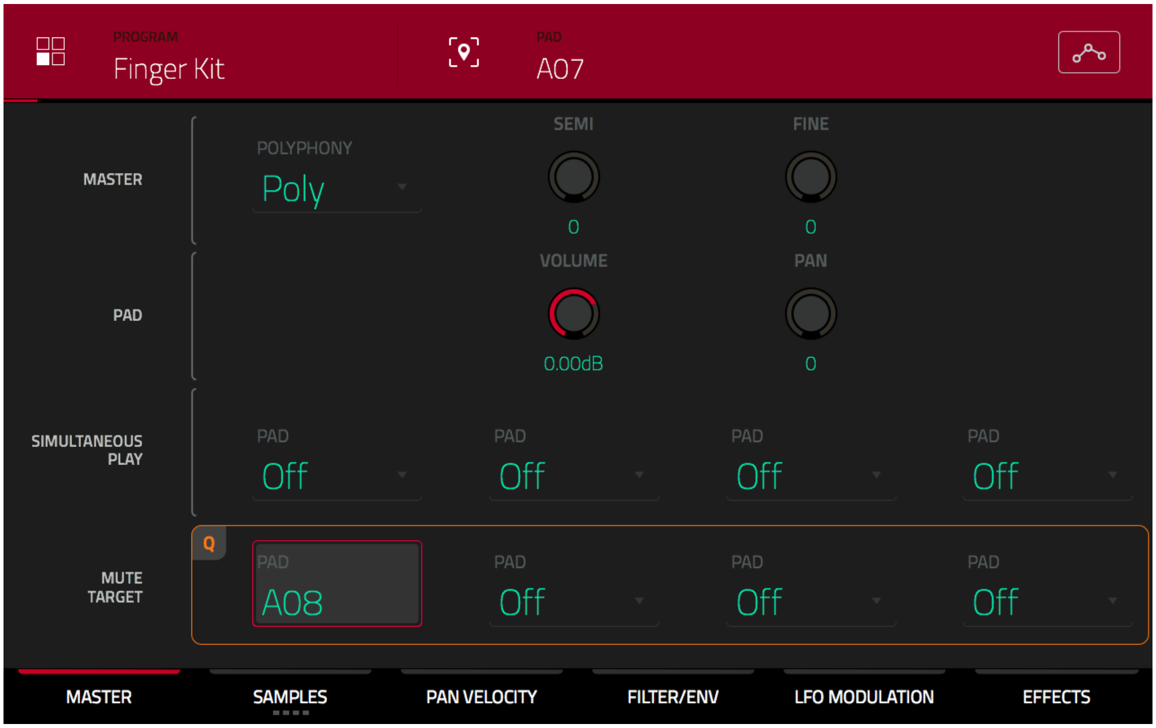
Muting the Open Hat with Mute Targets

We previously looked at muting an open hi hat via 'mute groups'. DRUM programs offer us a slightly more refined muting option in the form of M**UTE TARGETS**.

The advantage of mute targets is that they allow us to set up a *one way mute*. Select pad **A07** and go to the **MASTER** screen; look at the bottom row:



Here we can set up to four mute targets for our closed hat. Any pad number we enter here will be muted by our A07 closed hat. Set the first mute target to '**A08'**:



That's it! Hit **A08** quickly followed by **A07** and the mute works perfectly.

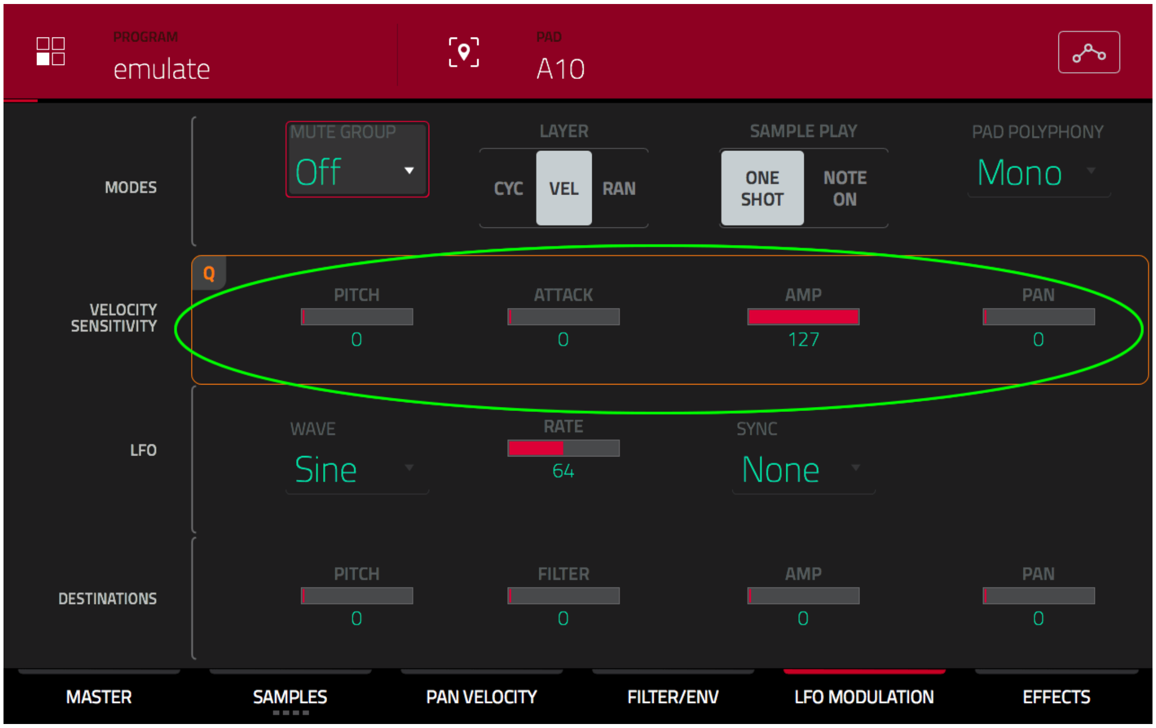
If you want A07 to mute other sounds, you can add up to three further mute targets. However, that's the maximum, so in this respect mute groups win hands down as in theory you could assign an entire program to the same mute group so all 128 sounds could mute each other.

Load up my version of the kit; **Finger Kit Vel-Cyc.xpm**.

Velocity Sensitive Timbre Emulations

It's not always possible to obtain multiple timbre variations of our drum sounds, but the MPC offers a number of ways to emulate these variations *on-the-fly*!

This is done using the built in 'velocity sensitivity' settings, which are found in the **LFO MODULATION** screen:



The idea here is that you can manipulate a specified program parameter based on how hard you hit the pad. To show exactly how this works, firstly turn off **FULL LEVEL**.

Now select pad **A10** and once again assign our hard snare, **RS_SN_H_1** to this pad. Start previewing pad **A10** by striking it at different velocities – try starting off very softly and gradually getting harder and harder. Obviously you will be able to hear the difference in volume, but that's the only changes you are going to hear.

V > ATTACK

Remember how I explained how softer hits have more attack taken off them? Well the **V > ATTACK** value is able to remove increasing amount of attack off your sample the *softer* you hit it. If you hit the pad at full velocity, it takes no attack off the snare giving you the full crack as you'd expect. Hit it with a 'medium' velocity and it will take off a 'medium' amount of attack.

Tap the '**ATTACK**' parameter and start turning the data wheel clockwise. The value you are setting now will determine the *maximum* amount of attack that will be taken off your snare – this is the attack setting that gets applied at the softest possible velocity you can strike a pad with – i.e. a velocity of '1'. Set this to an **ATTACK** of **65** .

Now repeat the soft-to-hard pad preview we performed earlier. This time, not only is the volume of the snare varying with the velocity (as you'd expect) but the *attack* at the beginning of the snare is clearly changing depending on how soft or hard your hit is. The softer hits clearly have a lot less attack left on them, while the hard hits are nice and punchy.

V > PITCH

We can do the same with the tuning of our sample using the **V>PITCH** setting. Set this to something pretty high, such as **80** and once again play pad **A10** at varying velocities – you should hear the softer notes have little or no tuning changes, while your hard hits have a very high pitched tuning. Obviously this is an extreme example and we need to be more subtle, so reduce the **PITCH** setting to **6**. You can now hear that as you vary the velocity, there are very subtle changes to the tuning of the sample.

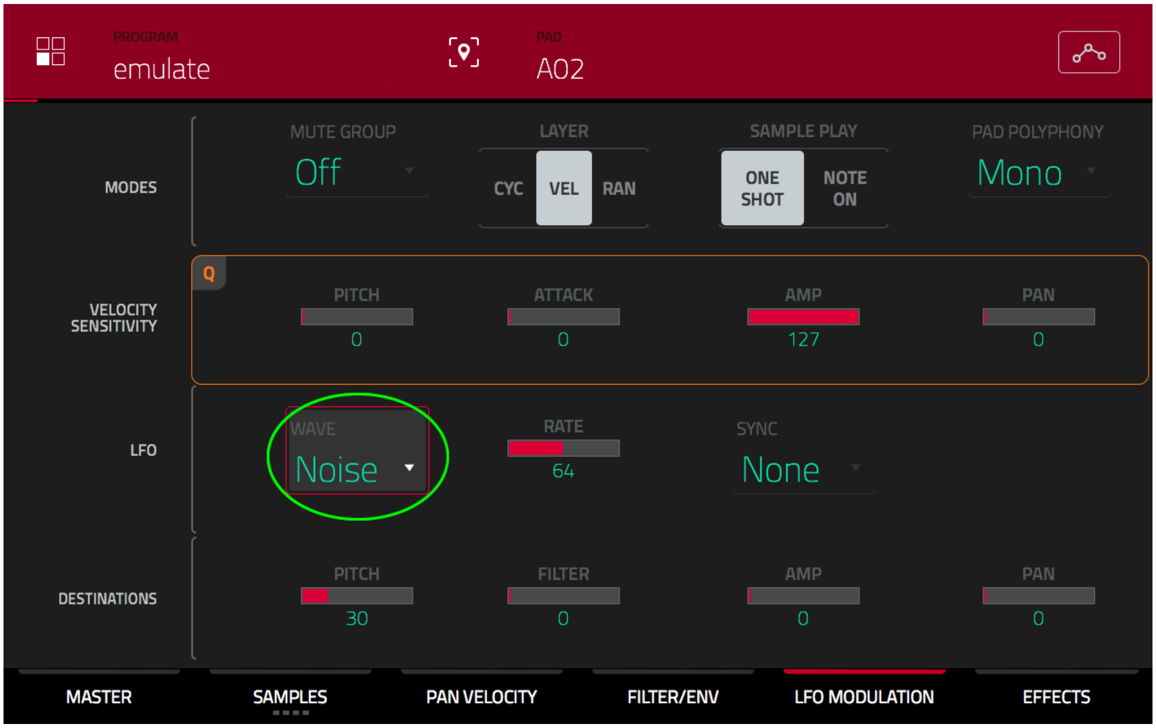
V > PAN?

This one is less about realistic timbre emulation and more about creating an interesting effect.

With **PAN** set to **127** you can hear it in its full extreme glory – simply strike **A10** multiple times starting soft and gradually increasing to full velocity and you'll hear the snare move from the left to the right of the stereo field.

Utilizing The LFO

One very interesting timbre emulation parameter can be utilised through the Low Frequency Oscillator (LFO). First, select pad **A02** and assign the hard snare **RS_SN_H1** sample to **layer 1**. Now go to **LFO MODULATION:**



On the **LFO** row, set the **WAVE** parameter to **Noise**. This setting generates random modulation and can actually create very realistic timbre variations. Currently it is having no effect, but tap the **PITCH** setting on the **DESTINATIONS** row, change this to **70** and keep hitting pad **A02** continuously to hear the result. You should hear that the random LFO is quite obviously affecting the timbre of the snare and in my opinion this sounds fairly natural.

I reckon a setting of **30** is probably the best here, it's not too obvious, but it still generates some interestingly different (but subtle) timbres.

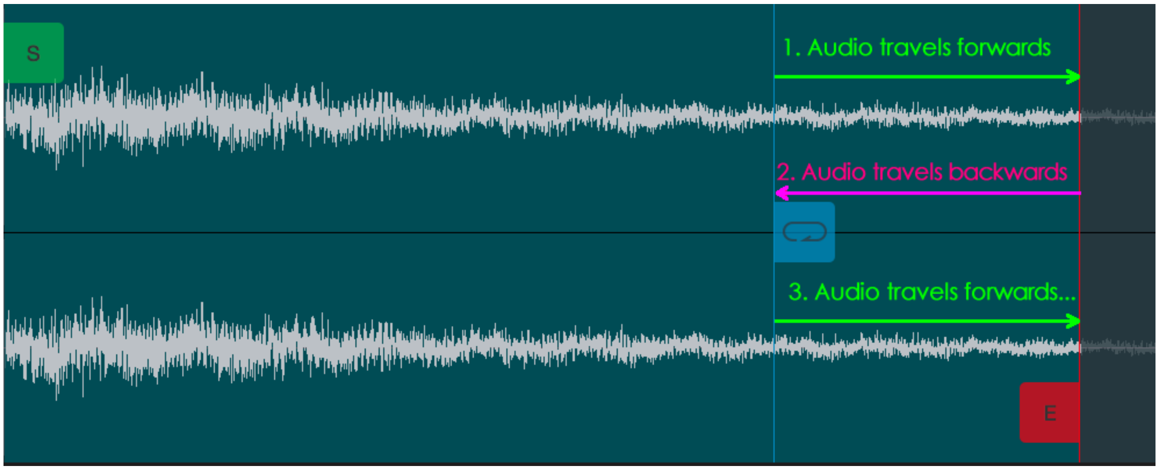
Extending the Open Hats

Currently our chopped open hat is fairly short so it would be nice if we could lengthen it, especially if we are playing a slow tempo beat where open hats can often stay 'open' for longer periods.

Method 1: ALT Looping

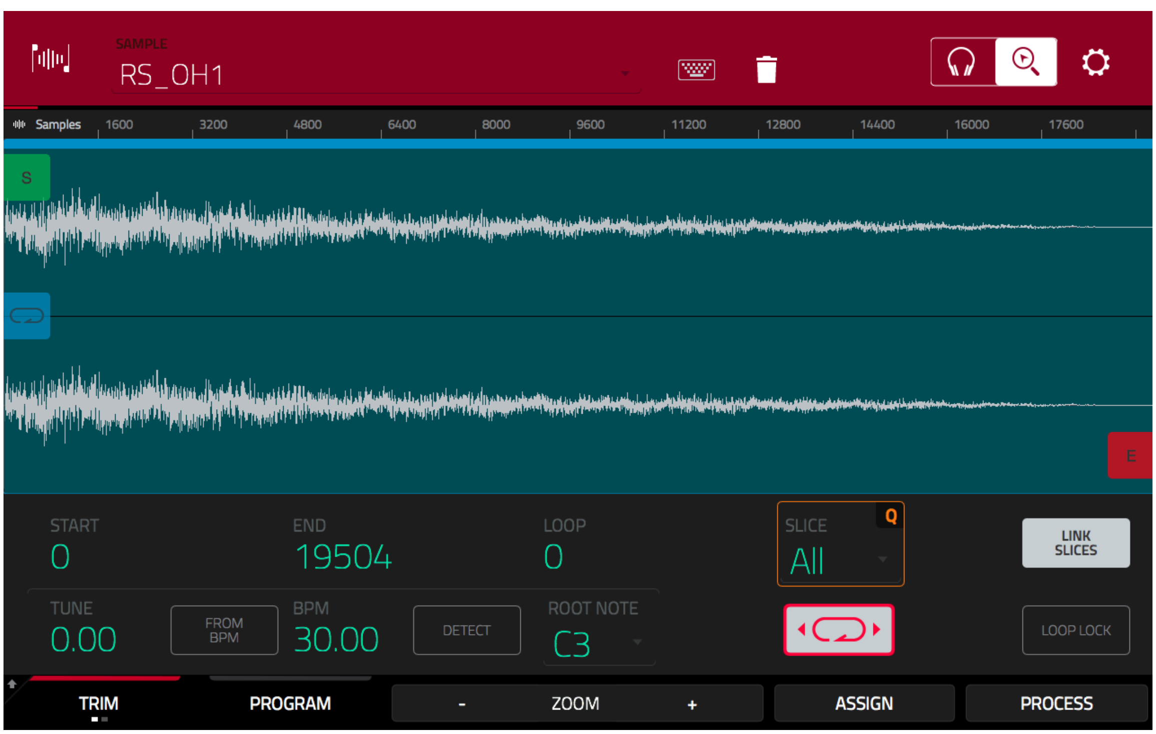
We discovered looping in chapter 014, where we looked at standard 'forward' looping to create sustained loops. A sustained loop can make sense when dealing with open hats as it gives us the option of leaving the hi hat 'open' for as long as we require.

A quicker way of sustain looping which works well with percussive sounds is 'alternating' looping, or **ALT loop** for short. With an alternating loop when the audio reaches the end point it doesn't jump instantly to the loop start point, instead it travels *in reverse* back to the loop point. Upon reaching the loop start point it travels forwards again towards the end point:



This alternating playback means that there's never a harsh, sudden jump from end point to loop start point, therefore you tend to get smoother transitions at the loop points. The downside is that half the loop is played in reverse which can sound odd with musical notes, but is often not noticeable with percussion sounds.

Select the **RS_OH1** sample. Turn off **LOOP LOCK** and set the **LOOP** to '**ALT**' type:



Tap the **LOOP** point and move it to an area where the open hat's waveform is becoming more consistent, around **10000**. Now tap the **END** point and move this back to a position before the hat starts decaying, around **14000**.

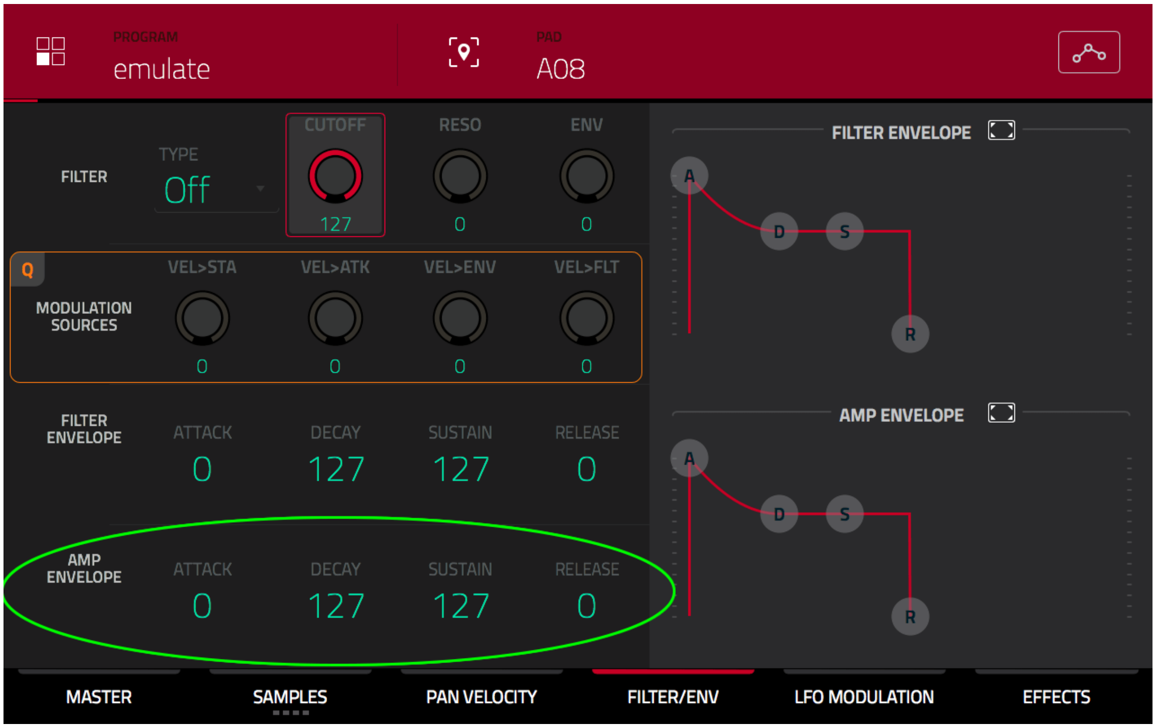
The best preview pad here would **be pad 11** 'Play Sample (Note on)' – this will play the sample from the beginning and then loop it for as long as we hold down the pad, mimicking the way we'll actually use the pad in our kit.

With **pad 11** held down, begin to adjust your **loop** and **end** point until you get a loop region that sounds pretty decent. Use **SNAP 0** (hold down SHIFT to reveal the sub menu to select SNAP 0) and **ZOOM+** (or pinch and zoom) to help automatically locate zero crossing points. I settled on '**LOOP: 10520'** and '**END: 13315':**

Go to **PROGRAM EDIT** and select **pad A08** of our kit. Now remember that as this sample is looped we'll first need to set our pad to **NOTE ON** playback in **LFO MODULATION** – if hold down the open hat it will now play back for as long as we wish.

The problem you might find is that when finger drumming, you might not feel comfortable having to hold down a drum pad as you'll probably be used to triggering them 'one shot' style. If you forget that the open hat is set to note on you'll end up with a very short open hat!

Luckily there is a pretty good workaround for this. Head over to **FILTER/ENV** and you'll see the **AMP ENVELOPE** has changed from standard AD to an **ADSR** envelope; attack, decay, sustain and release.



We'll take a closer look at this new envelope later in the book, but for the moment, just set a nice long **release** of **100**. Set a sustain of **127**. Just tap the pad and you hear it carry on playing (thanks to the combination of release and looping). And thanks to the mute target we set for the hat on A7, we can shut off our hi hat whenever we please by hitting A7.

Obviously this open hat might sound odd if you leave it playing for too long, but in your average drum kit performance it will work very nicely indeed – in fact if you listen to a real open hat they often 'sizzle' during decay and the alternate loop also emulates this a little. Try it out – load up my version from the **Kits** subfolder; **Finger Kit ALT.xpm**.

Method 2: Time stretch

Another way to extend the length of any sample is to time stretch it. Go to **SAMPLE EDIT** and select the **RS_OH2** sample (this was loaded into our project at the start of this chapter). Press **PROCESS** and select **Time Stretch**. Now, this is a single hit so tempo isn't relevant, but, regardless set the **BEAT** (number of beats) to **1**.

Now what we want to do is make this sample twice as long – to do this, use your cursor to highlight the **RATIO** field and set it to **50.0**. Hit **DO IT** and have a listen

– it's not bad, perhaps slightly degraded at the beginning, so let's try a smaller stretch. Press **UNDO** to return to the original un stretched sample and press **PROCESS** again. This time try a '**RATIO'** value of **60.0**. Press **DO IT** and preview the result. Sounds good to me!

You can of course keep on experimenting – just remember to UNDO after each attempt, because 're-timestretching' over the same sample will just increase the nasty artefacts.

Load up my version from the **Kits** subfolder; '**Finger Kit Stretched.xpm'** – this contains our stretched open hat on pad A08; this time I've left the pad as a 'one shot'.

A Practical Kit Set Up

Load up the project file '**finger drumming break.xpj'** from the **chapter 017** folder and select sequence 1. On track 1 I have assigned my final version of the kit, **Finger Kit Final.xpj**.

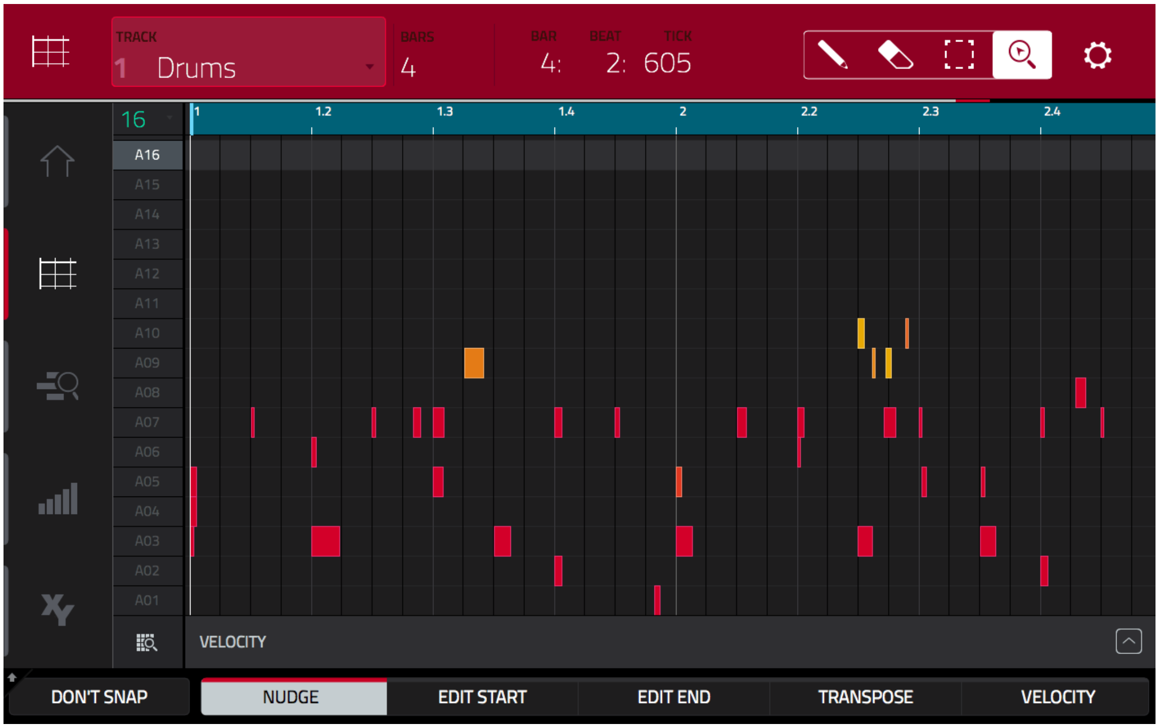
Here I have tweaked the kit here and there to help improve its performance for finger drumming. Basically every technique we've covered in this chapter is there in some form or another, but I've added additional pad assignments to help with common finger drumming requirements.

For example I have also set up closed hats on pad **A03** so if I wish, I can perform some more intricate hat patterns using two fingers on my right hand.

I also added a single crash sample to pad **A04**. This wasn't chopped from a break, instead I searched through my drum collection to find a crash that suited the vibe of the kit I've built – this one has velocity sensitive emulation and a '**Noise' LFO** added.

Using 'Humanize' on Drum Performances

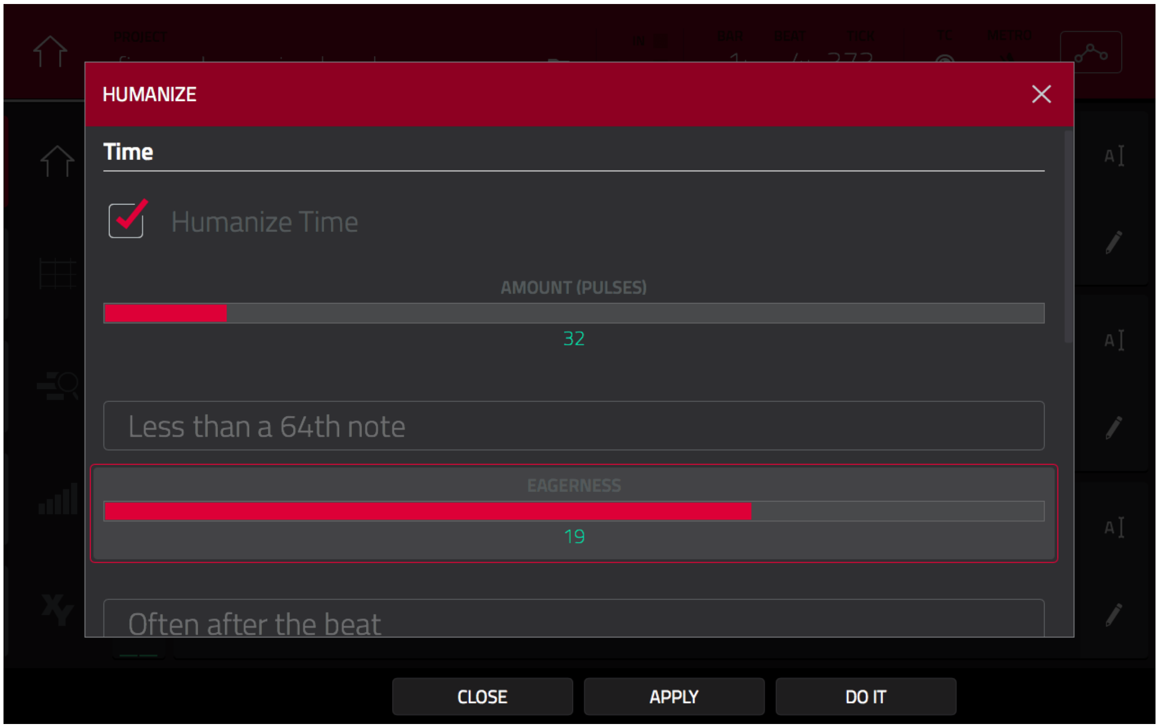
Sequence 1 (**Break - Initial)** features a sequence I wrote using the finger drumming kit; all events are recorded to track 1. Take a look at it in **GRID**:



Most of the sequence was written with a 1/16 quantise and FULL LEVEL 'ON' except for the ghost drum rolls/flams on pads A09 and A10.

We used HUMANIZE back in chapter 11 to give a more natural feel to our piano parts, but we can use it on drum tracks as well.

With **TRACK 1** selected in **MAIN**, hit the **pencil** icon on the **TRACK** row and select **HUMANIZE**:

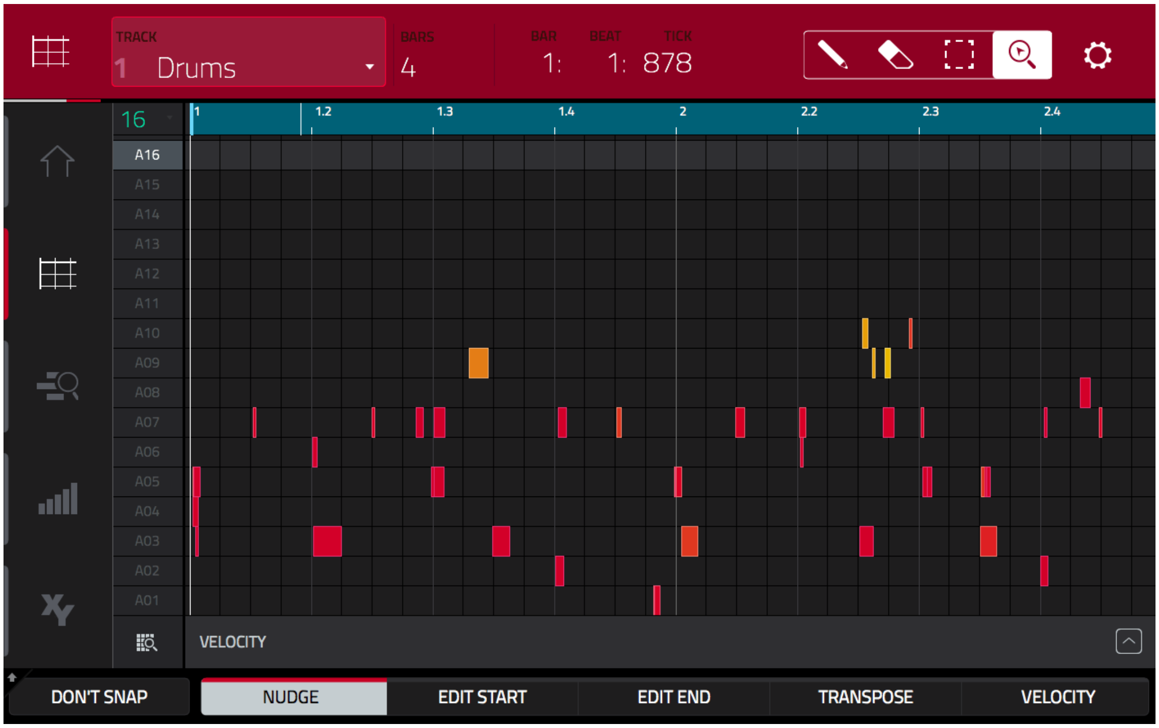


When it comes to humanising the timing of the beat, let's try some subtle 'after the beat' changes with an **EAGERNESS** of **19** and **AMOUNT** of **32**.

There's no point humanising note length as all our drum events are one shots.

Velocity is definitely one aspect that will benefit greatly from humanising, so set this to a mid value, around **40**.

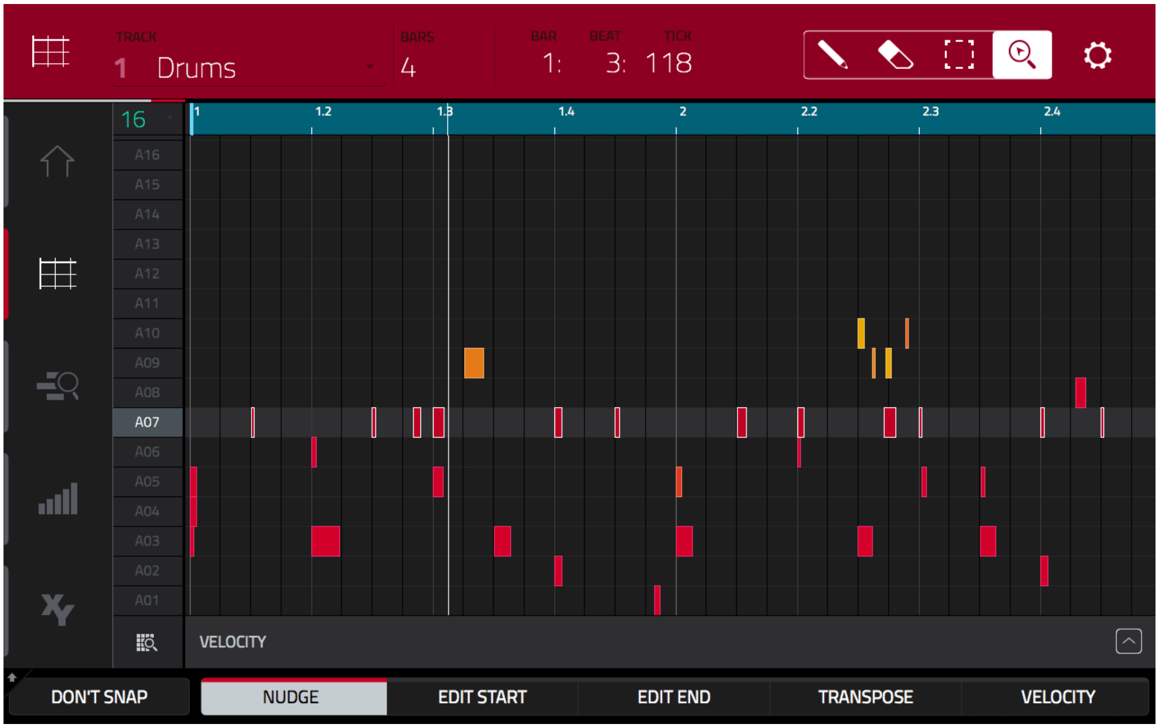
Finally let's leave the '**Input Filter**' unchecked as in this example I'm going to humanize all events together. Hit **APPLY** and **PLAY START** to hear the difference. Alternatively, select **sequence 2** (**Humanize - All Events**) to hear my version.



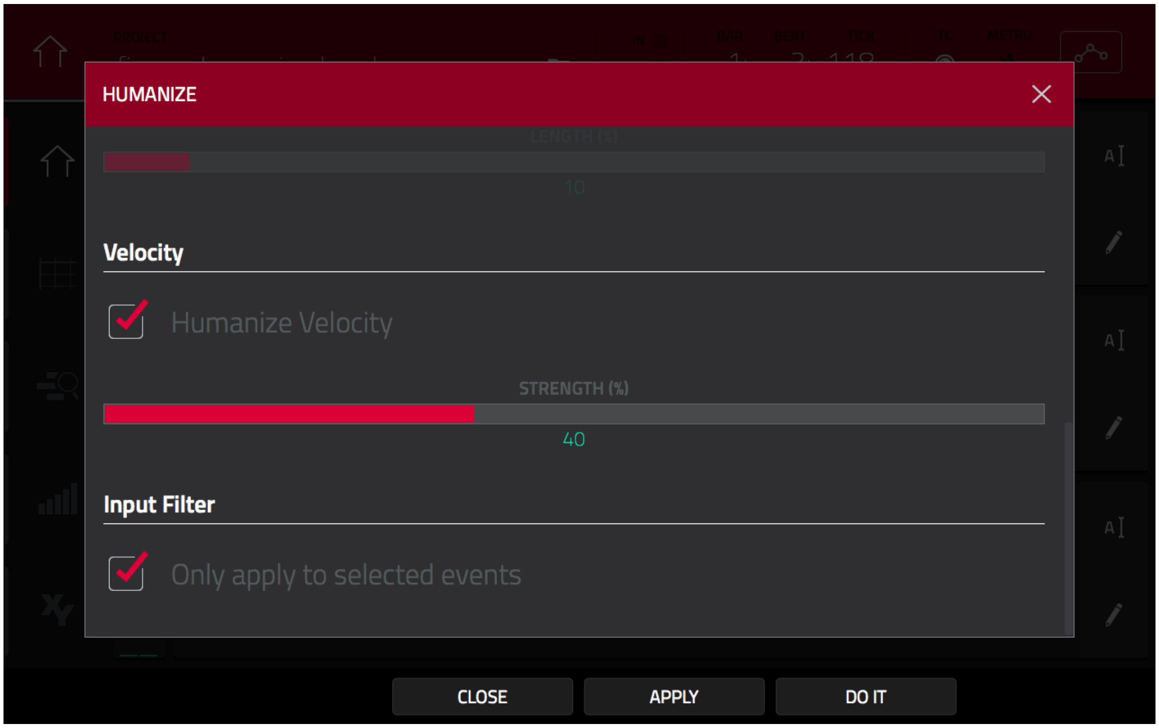
I think some of the resulting sequence sounds pretty good, especially the first two bars, but bars 3 and 4 don't quite work, they now sound too sloppy.

You can of course continue to experiment with your humanize settings, or you could manually edit the problematic notes. Another option is to break things down by instrument type and humanize each one separately.

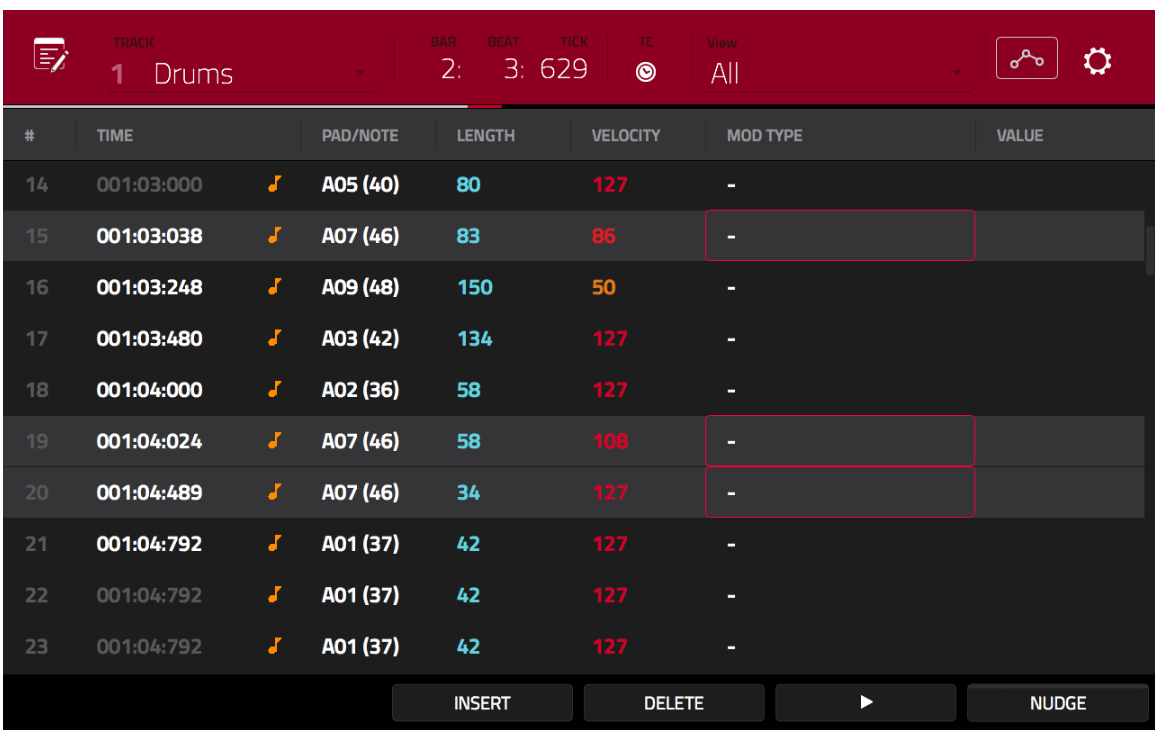
Hit **UNDO** to bring sequence 1 back to its original state and in **GRID EDIT** select all the pad **A07** (closed hat) events. Remember if you've set **HITTING PAD SELECTS ALL EVENTS** to **ON** (hit the gear icon, top right of GRID screen), then hitting A07 will select them all for you automatically.



Go back to **MAIN > TRACK row pencil icon > HUMANIZE** and use the same settings as before (they should all still be configured by default). However this time at the bottom of the screen (remember to drag the page upwards) make sure you check the **Input Filter** so the humanisation is only applied to the selected hi hats.



Hit **APPLY** and have a listen to the result via **PLAY START**. If you are happy, hit **CLOSE** and go to **MENU > LIST EDIT**:



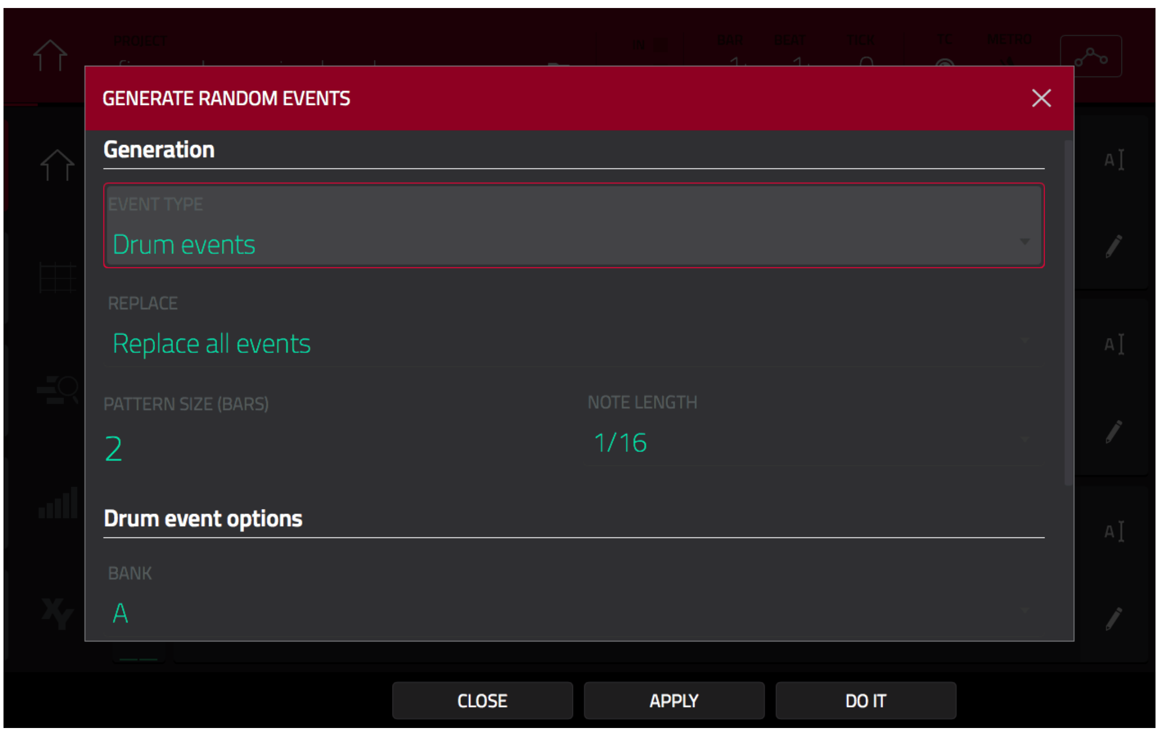
As you can see, the A07 events are no longer on exact 1/16th quantise points, while the velocity of the events now vary a little - in the example above, we have velocities from 86 to 127. Most importantly, all other events remain untouched.

You can now, if you wish, repeat HUMANIZE separately for other pads. For example, try humanising the main snare on pad **A05** - as this is a powerful core part of the beat, I'd advise keeping the snare at full 127 velocity, so try humanising the timing only.

Listen to **sequence 3** (**Individual Humanize**) - here I've applied some individual humanisation to many of the drums, including some velocity humanisation on the kicks; it's definitely more natural sounding compared to the original sequence 1.

Generating Random Drum Events

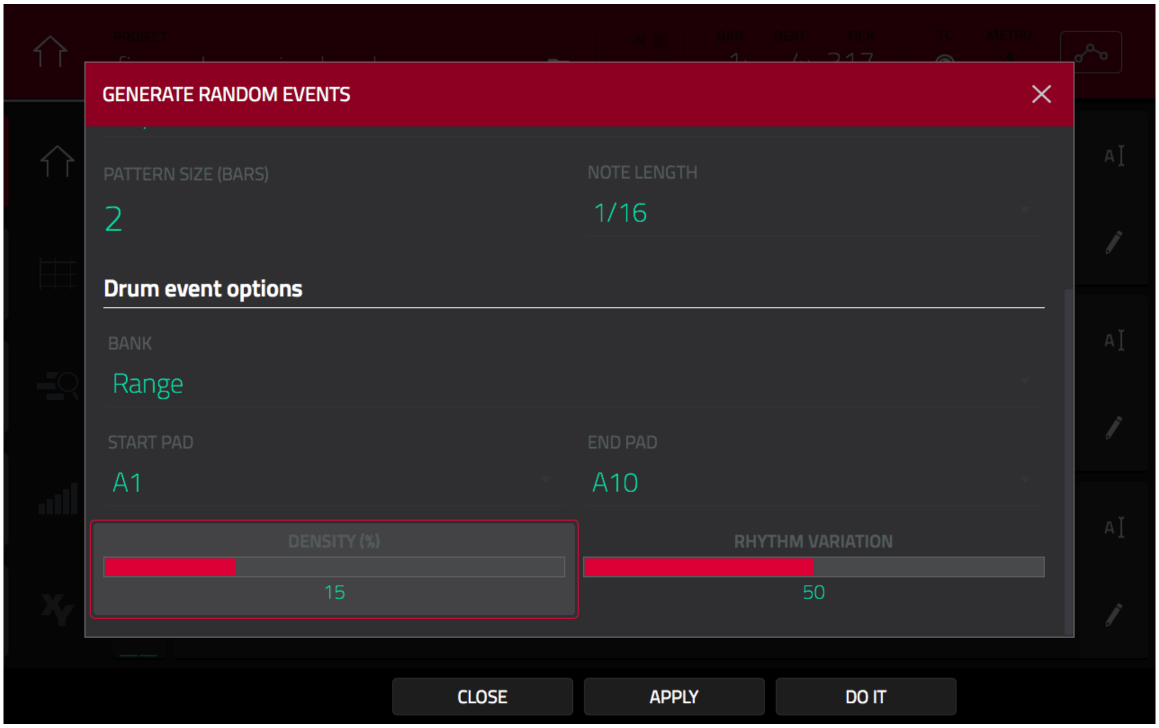
If you're really stuck for ideas for new drum tracks, let the MPC generate some drums for you! Select **sequence 4**, which is just an empty sequence and on **track 1** go to **TRACK pencil icon > GENERATE RANDOM EVENTS**:



We used the feature back in chapter 11 but that was for generating melodic events. This time as we are using a DRUM program, the MPC defaults the **EVENT TYPE** to '**Drum Events**'.

As we discovered previously, the **NOTE LENGTH** setting seems to indicate the *quantise* used in the random generation, and I would advise a setting of **1/16** is always a good starting point.

For **DRUM EVENT OPTIONS**, select the **Range** option:



Set the **range** to cover all the active pads in our kit, **A01** to **A10**.

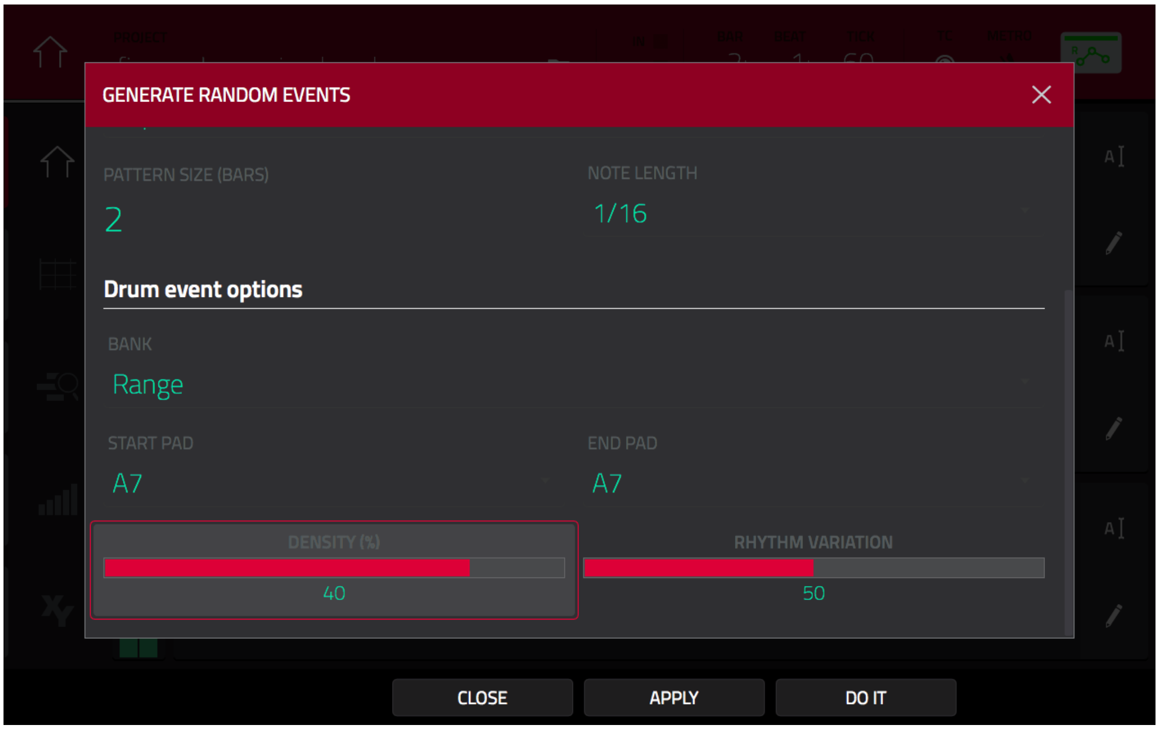
Set the **DENSITY** quite low, otherwise we'll flood the track with far too many events, try an initial value of **15%**.

Set a medium **RHYTHM VARIATION** of **50**. Hit **APPLY** and **PLAY START** - have a listen and if you don't hear anything useful, keep on hitting APPLY until you do - and remember to adjust the DENSITY and RHYTHM VARIATION occasional to see if they help. You can hear one of the random generations I made on **sequence 5**.

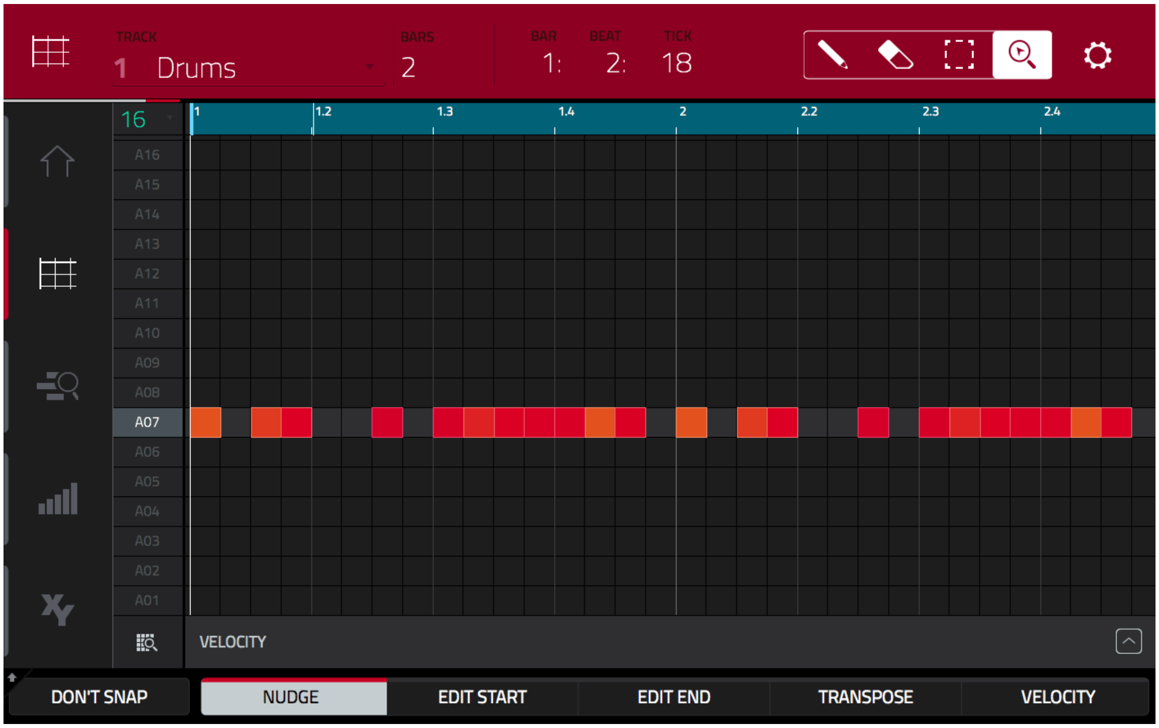
Now it is incredibly unlikely that you'll generate the perfect break this way, but you might find something worth isolating from your sequence. It might be a little fill or a partial groove. When you do find something, just break out all the editing functions you've learnt previously in the book and begin building.

Randomly Generating Hi Hat Patterns

I do however feel that random generation works much better when used on a single pad. Go back to the **GENERATE RANDOM EVENTS** screen and this time select a range covering only pad **A07** (a closed hat):



This time set a higher density as we are only using a single pad and want to generate a full hi hat pattern. Hit **PLAY START** and begin hitting **APPLY** until something sounds good. After a few attempts I got a great hat patter, which you can hear on **sequence 6** (**Random Hat**):

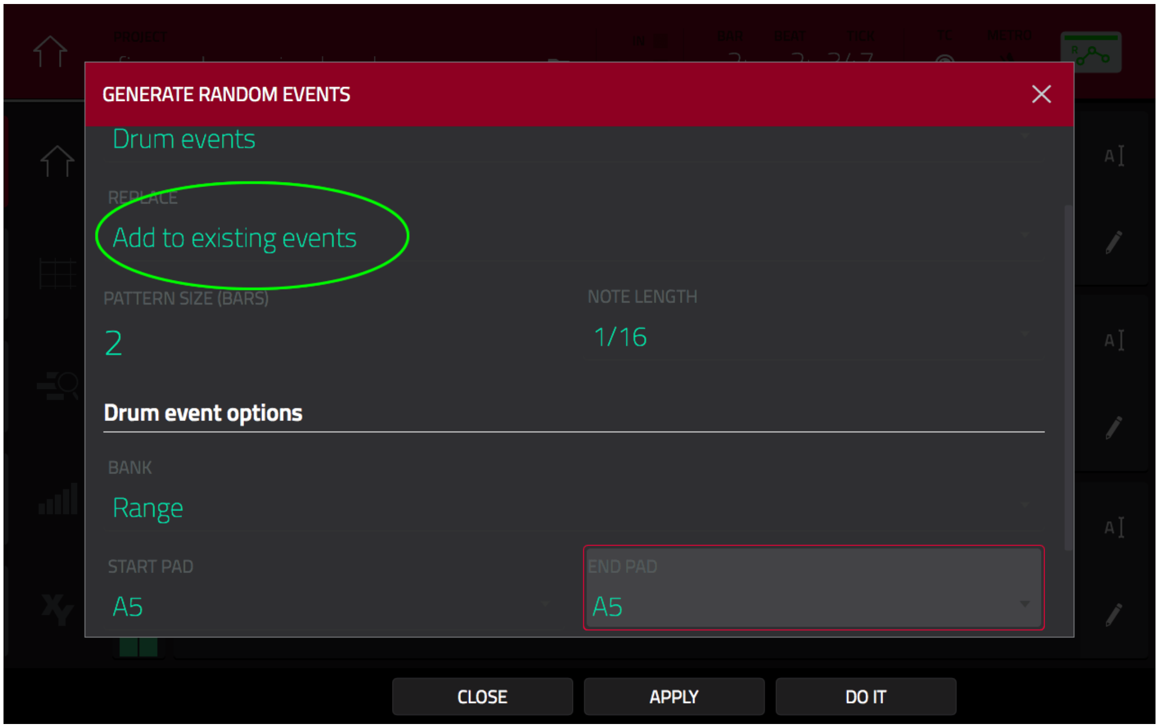


At this point you could hit **OVERDUB and PLAY START** and start manually adding some kicks and snares in real time - check out **sequence 7**.

Individually Generating All Drum Parts

Another option is to continue to randomly generate individual instrument and there's two ways you could do this. Firstly you could isolate each instrument on its own track - for example, select **track 2**, go to **GENERATE RANDOM EVENTS**, choose a **range** of **A5-A5** and start generating some kicks (remember to experiment with density). Then go to **track 3** and generate some snares using a range A6-A6. Check out my version on **sequence 8** (**Random Tracks**) .

The second option is to keep everything on a single track but change the **REPLACE** parameter so it no longer wipes all events each time you hit APPLY. There's two options here; '**Add to Existing Events**' and '**Replace Events in note range**'.



With '**Add to Existing Events**' selected, each time you hit APPLY the MPC will generate more random events to the track, leaving the existing hat events in place. In the above example, I've set the '**Range**' to **A5-A5**, which will generate A5 kick drum events each time I hit APPLY.

'**Replace Events in Note Range**' will also generate new events, however it should replace the notes selected in the 'Range' section - so, with A5-A5 selected, it should just keep replacing the existing kick pattern with new kick events while leaving the existing hats in place. The problem is that as far as I can see, this feature is not working correctly at the moment, so I will update this tutorial when/if Akai fix it.

In the meantime, either use the individual track method, or use 'Add to Existing Events' and if you don't like the result hit UNDO to remove the generated events and hit **APPLY** again; repeat until you've generated a cool kick pattern to complement your hats.

Select **sequence 9** for my 'individually' generated sequence.

So lots of options - random generation really is a great way to get out of a creative rut, I especially like using it for hat patterns. Remember everything can be tweaked in GRID EDIT or via real time recording or any other technique I've covered elsewhere in the book - and while you're playing with randomisation, don't forget you can still apply some additional HUMANIZATION to the resulting sequences.

018 Dynamic Tempo Manipulation

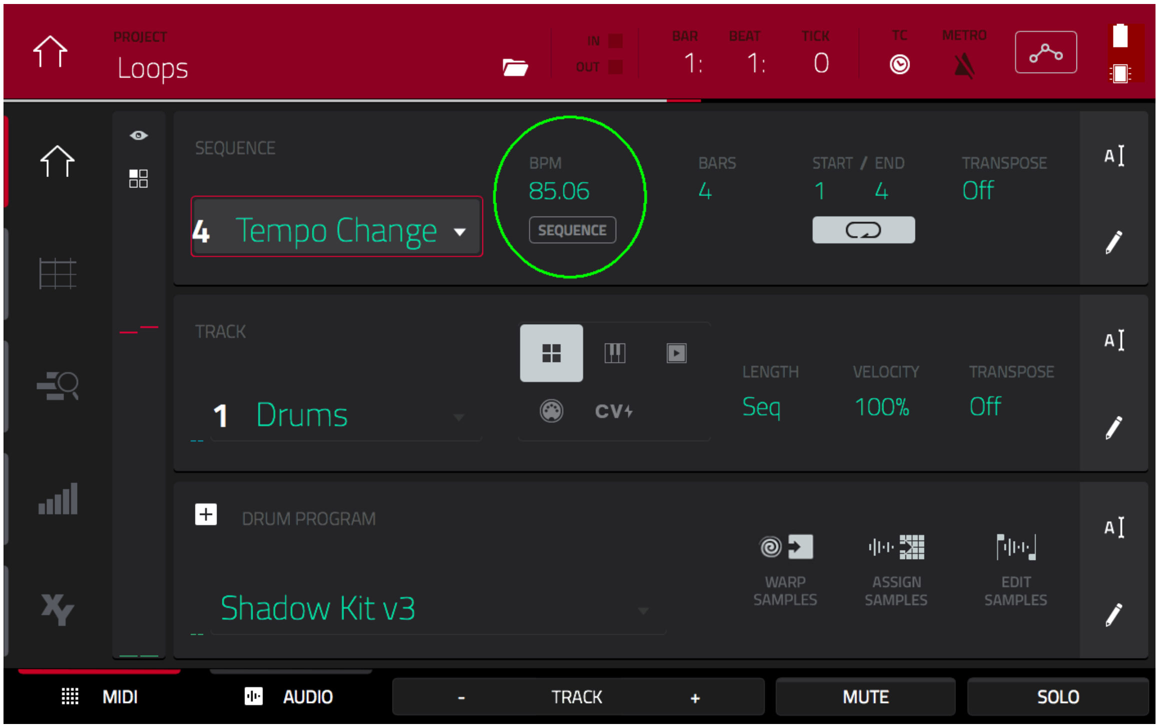
In our Looping tutorial we tuned and timestretched the cowbell loop so it would match the tempo of our existing 'Demo Beat' sequence. However in the MPC we can also dynamically change the tempo of audio to automatically adapt to tempo changes.

Problems When Changing Sequence Tempo

Skip back to the **'014 Working With Loops'** folder and load up the **Loops.xpj** project we were working on previously. The **Demo Beat** sequence that makes up this project has a fixed **tempo** of **90 BPM** which meant that our **85.06 BPM** cowbell loop (**old-cowbell-looped**) had to be tuned or timestretched to 90 BPM before it could be used within the sequence.

There are however other options when it comes to tempo matching samples and sequences. One option is to *change the sequence tempo* to match that of the loop.

Go to **sequence 4 (Tempo Change)** of the **Loops.xpj** project – here I've assigned the original (85.06. BPM) old-cowbell-looped sample to track 6, but most importantly I've changed the **tempo** of the entire sequence from 90.0 to **85.06**.



Press **PLAY START** and listen to the slower version of the beat. Doesn't sound so great does it? However, if you **MUTE** the two audio tracks then the sequence sounds fine. What's the problem?

The problem is that these audio tracks were recorded at the original 90 BPM so they cannot be played in an 85.06 BPM sequence. However any track made from single 'hits' and individually sequenced MIDI events, such as the drum track and bass line will happily play at the new tempo, in fact they'll happily adapt to any tempo; that's one of the big benefits of using MIDI vs audio tracks.

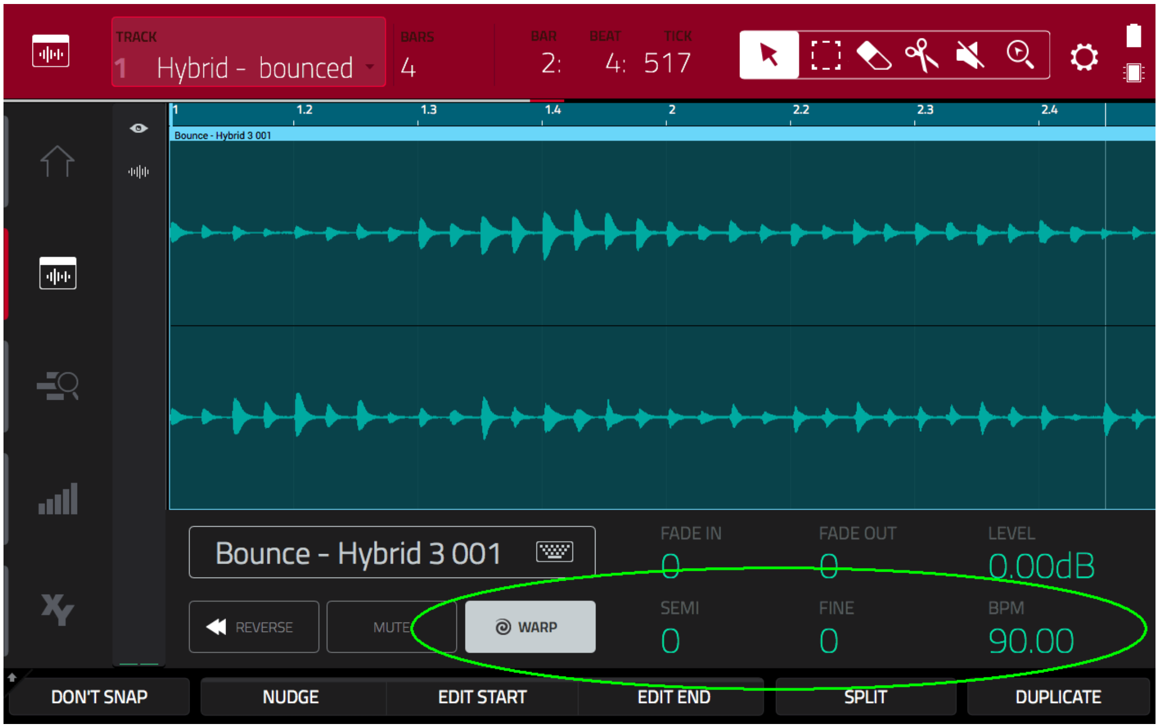
All is not lost though because the MPC has a little trick up its sleeve; **Warping**.

Warping Audio

We've already discovered the ability to 'timestretch' audio within the SAMPLE EDIT screen. And in fact as audio tracks are nothing but 'samples' held in memory, we could actually open both our audio tracks in SAMPLE EDIT, hit PROCESS and timestretch them both manually from 90.0 to 85.06 BPM.

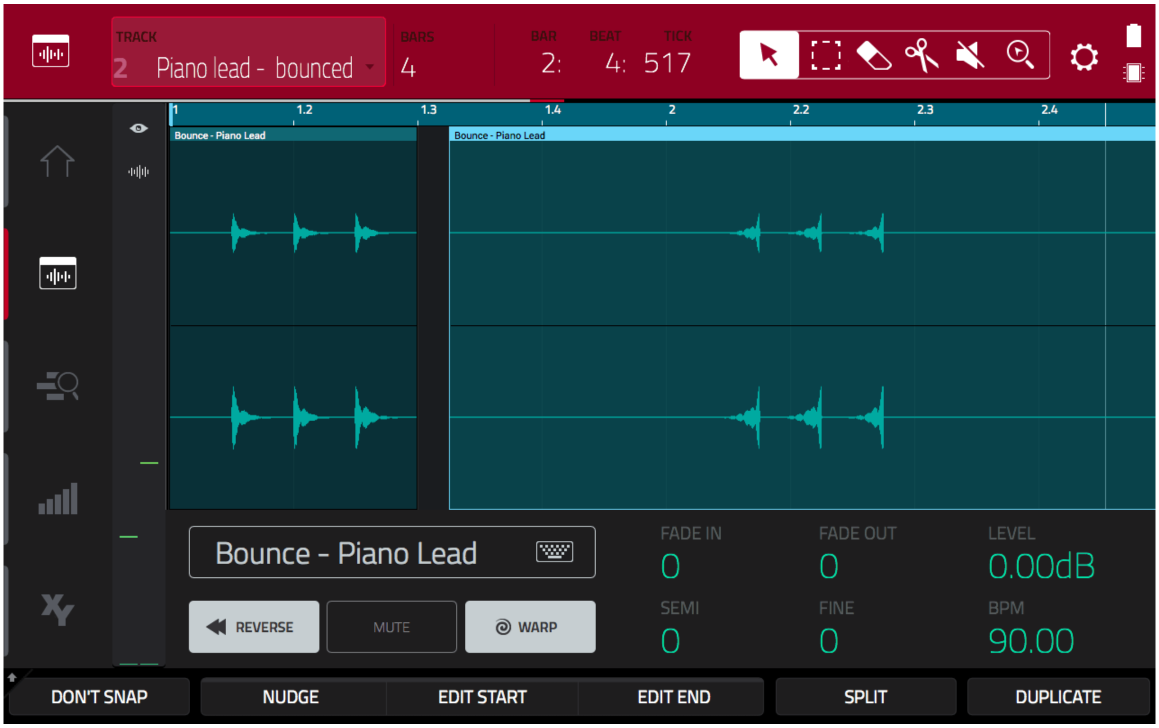
This is, for want of a better phrase, 'static timestretching'. Each time you want to change the sequence tempo you will have to head back to SAMPLE EDIT and re-timestretch your poor old audio tracks. It's doable, but it's not ideal.

Load up the project **Dynamic Tempo.xpj** from the **chapter 018** folder. From **MAIN** select **sequence 1**, **AUDIO** > **EDIT AUDIO** and choose **audio track 1** (**Hybrid bounced**). Select the **pointer tool** and tap on the waveform to select it; you'll see the **WARP** button at the bottom of the screen has been activated.



With warping activated, you'll now have access to the WARP configuration (circled). As you can see, this audio track is assigned a **BPM** of **90.0** which we know is correct as this was the tempo of the sequence when we originally bounced down the audio.

Select **audio track 2** (tap the audio track name at the top left of the screen and hit the + button). As this audio track is comprised of several small segments, you'll have to select each segment individually to activate warping for each segment.



With warping applied to all audio on both audio tracks, hit **PLAY START** to hear how they sound when played at 85.06 BPM. Sounds pretty awesome!

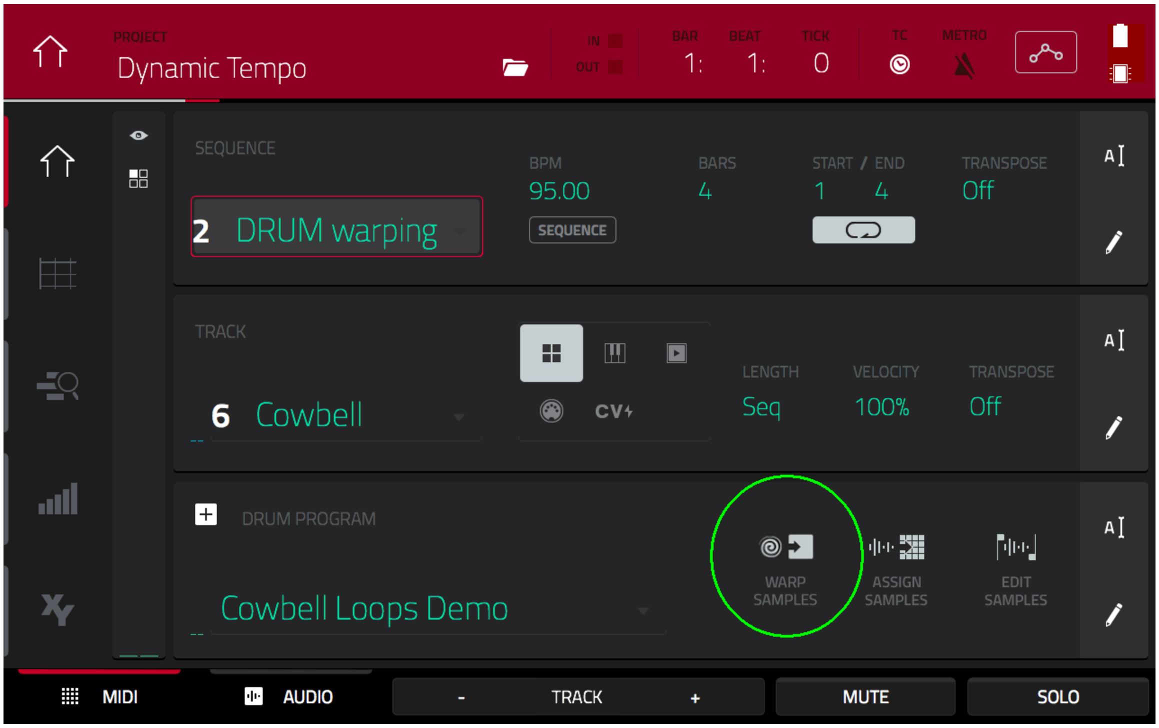
So what's happening? Well warping is basically a 'dynamic' form of time stretching; when an audio sample is 'warped' it automatically adjusts its tempo to match that of the sequence it is currently part of.

This means we can change the tempo of our sequence to any BPM we wish and the audio tracks will now automatically adapt to that tempo. The problem is that our '**old-cowbell-looped**' sample, which is located in our **Cowbell Loops** DRUM program, is fixed at 85.06 BPM, so this currently does not adapt to any other tempo automatically.

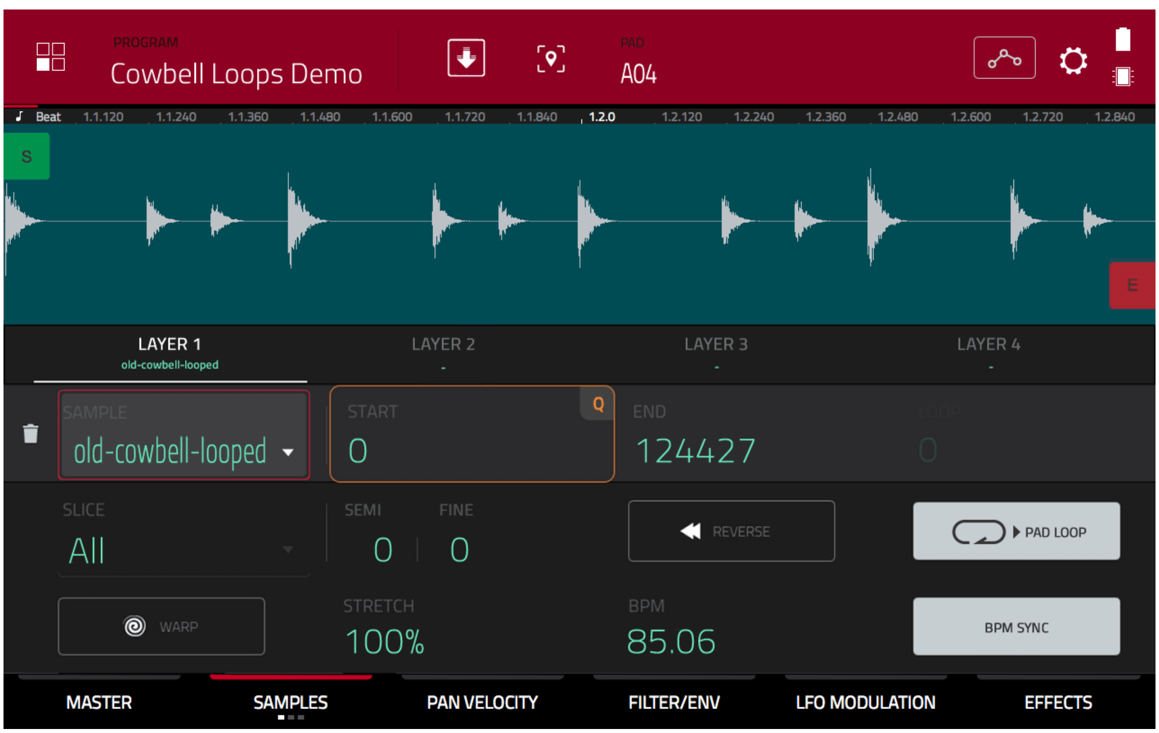
Warping in DRUM Programs

Select **sequence 2** of the current **Dynamic Tempo.xpj** project. Here I have set the sequence **tempo** to **95.0 BPM**. Press **PLAY START** – as you can hear, the 85.06 BPM cowbell loop on (MIDI) **track 6** ('**Cowbell**') is not in sync with the rest of the track. Let's use warping to fix this.

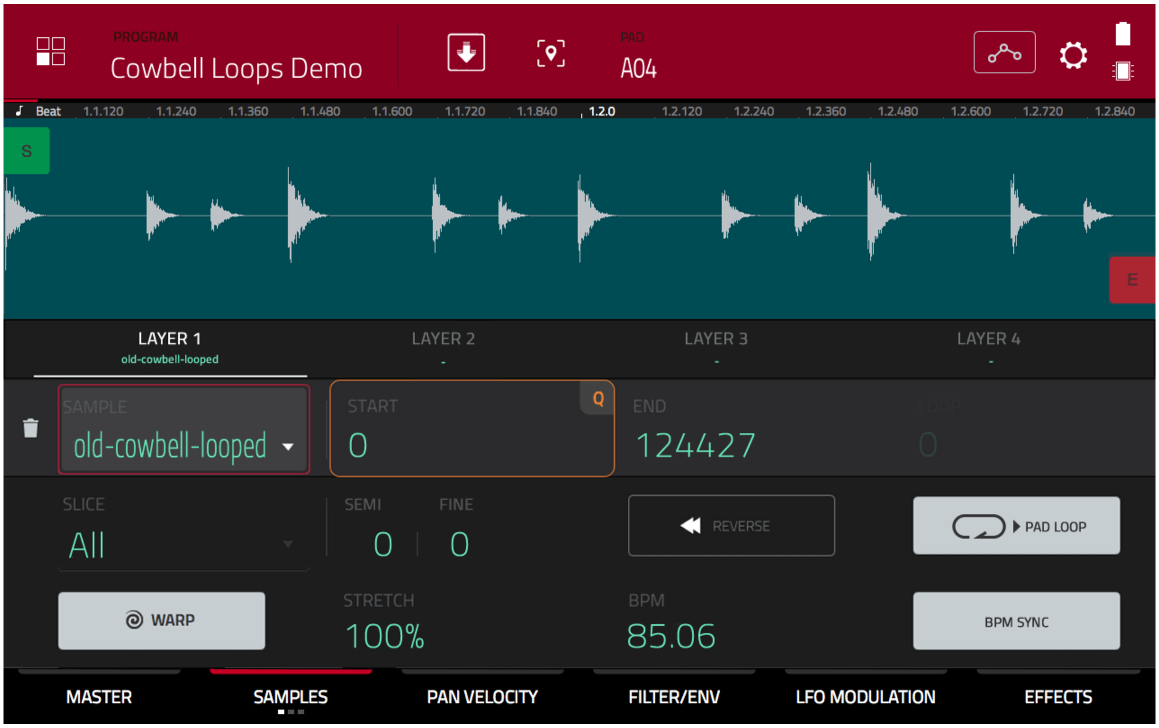
In **MAIN**, hit **pad A04** (which contains our '**old-cowbell-loope**d' sample) and select **track 6** (**Cowbell**). Now tap the **WARP SAMPLES** icon:



This takes us directly to **PROGRAM EDIT > SAMPLES**:



At the bottom of the screen you'll see the WARP settings. As you can see the original cowbell loop is correctly identified as having a **BPM** of **85.06 BPM**. Tap on the **WARP** button to activate warping on this sample.



Hit **PLAY START** to hear the cowbell loop now sync with the current sequence tempo - now change the sequence tempo to any value you wish and hear the cowbell adapt to the tempo in real time. check out **sequence 3** (**DRUM Warped**) where I've increased the tempo to 115 BPM.

BPM Not Displaying Correctly?

If the BPM is not showing at the correct tempo for your sample in PROGRAM EDIT, head over to **SAMPLE EDIT** and hit the BPM '**DETECT**' button - this should ensure the correct 'root' BPM is carried back to SAMPLE EDIT. Otherwise simply tap the BPM field in PROGRAM EDIT and manually set the correct BPM.

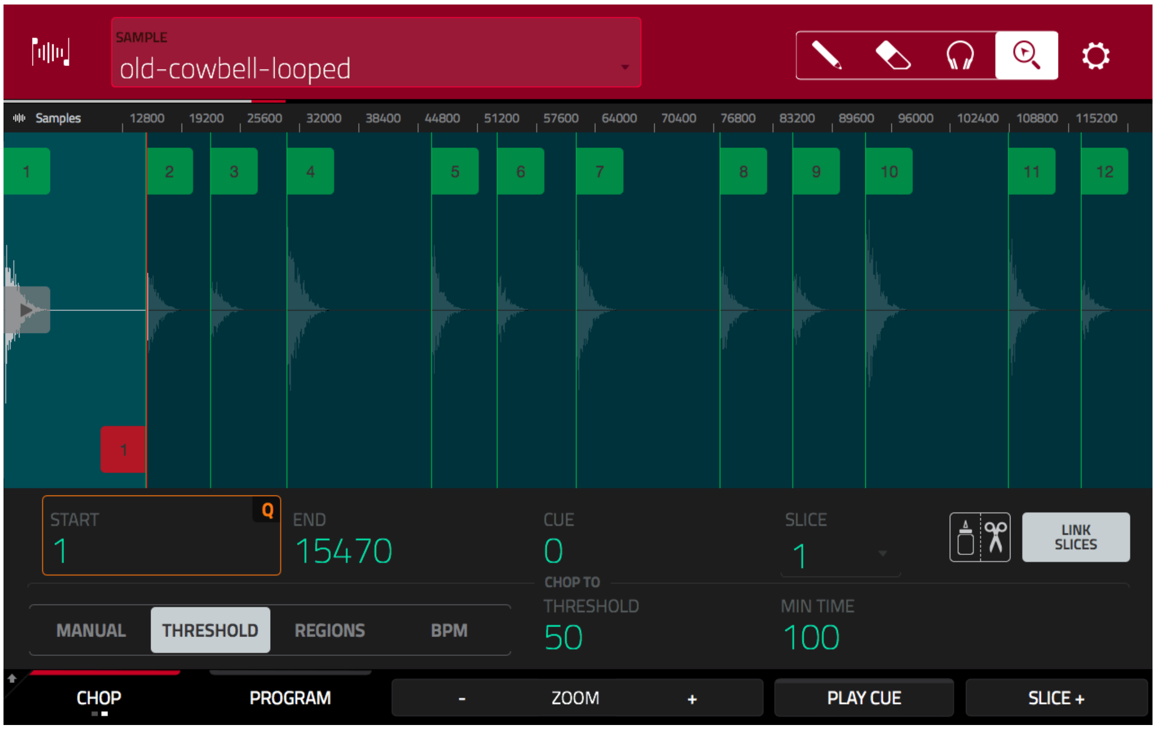
There's a few more 'warping' options here, but I'll take a look at these later in this chapter when we look at clip programs. Next up is yet another method of dynamically controlling loop tempo; patched phrase.

Patched Phrase Loops

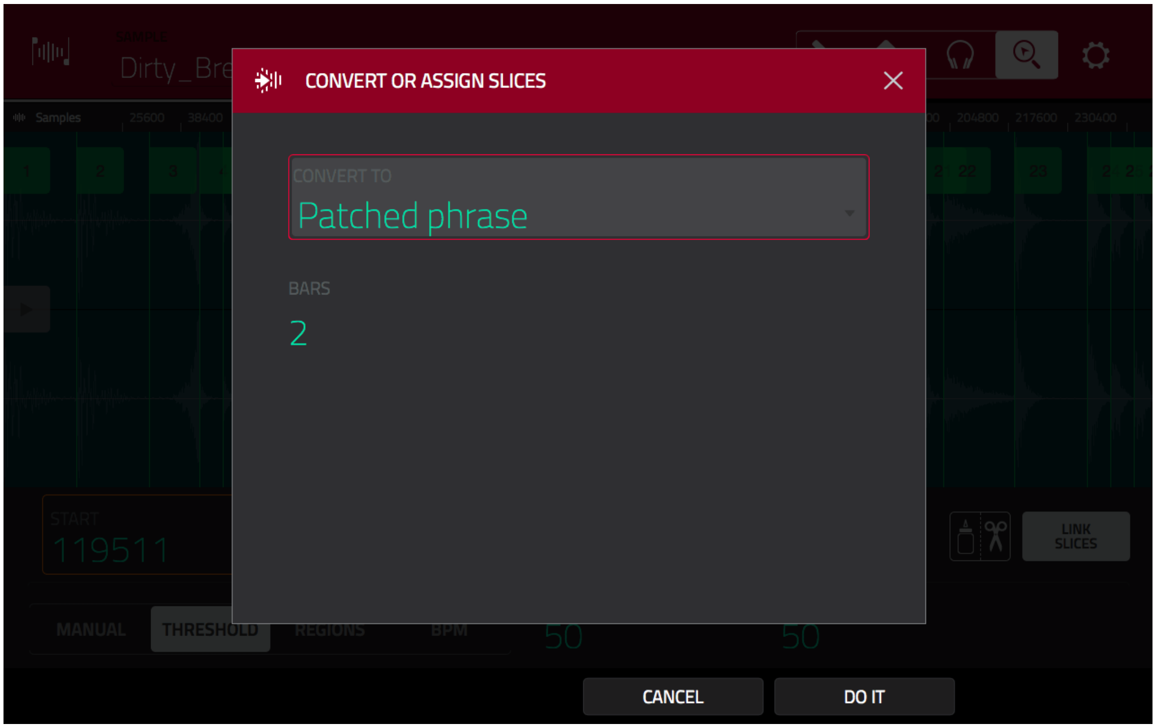
The MPC offers a cool alternative to warping in the shape of a **patched phrase**. A patched phrase is similar to a REX file or Apple Loop in as much as it's a loop dynamically adjusts to sequence tempo.

The beauty of patched phrases is they are very easy to create in an MPC thanks to the MPC's great threshold chopping feature.

Go to **SAMPLE EDIT > CHOP** and select the old-cowbell-looped sample. Choose **THRESHOLD** chopping with a **THRESHOLD** of **50** and **MIN TIME** of **100**:

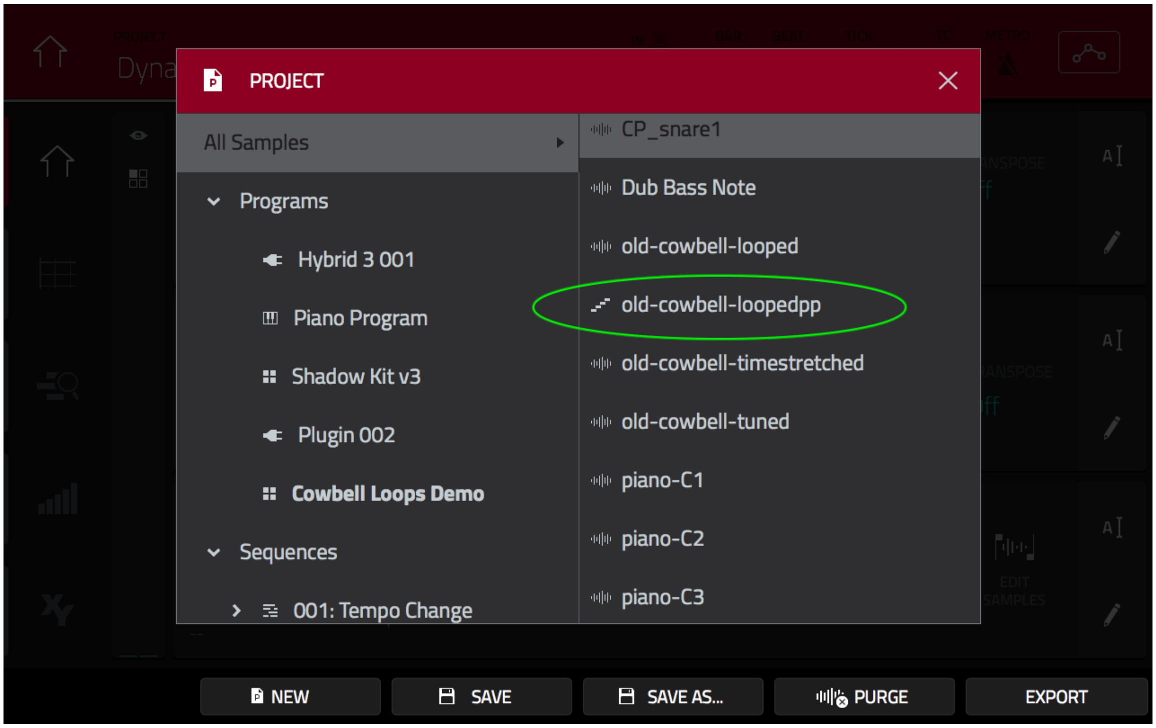


That's just about as perfect as a chopped loop can be. Hit '**SHIFT > CONVERT**', but this time from the '**Convert To**' field, select '**Patched Phrase**'.



As you can see, the MPC is asking us for the length of our loop – our cowbell loop is **1 bar** long (count 1, 2, 3 and 4) so set **BARS** to **1** and hit **DO IT**.

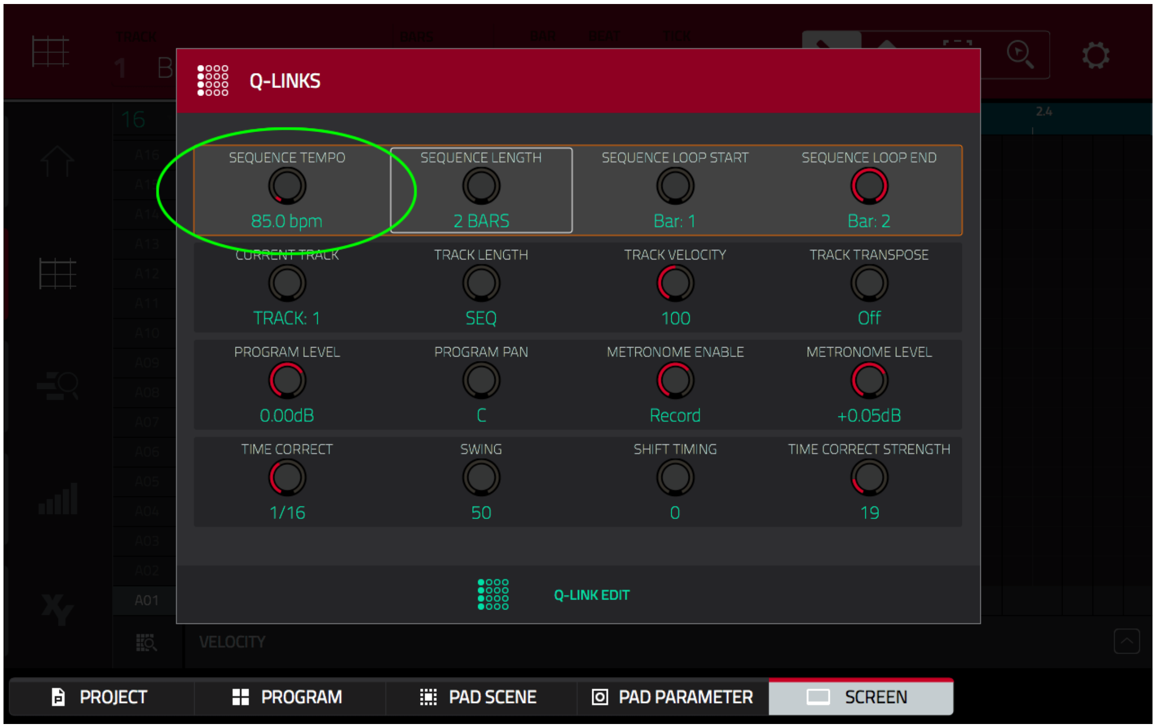
At this point it may not be particularly obvious as to what has happened, go to **MAIN** and bring up the **PROJECT** panel (tap the project name in the top left of the screen) and make sure **ALL SAMPLES** is selected – scroll down to the bottom of the list:



You'll see a new sample has appeared, **old-cowbell-loopedpp** – this is your *patched phrase* sample (note the 'pp' postfix).

In **MAIN** select sequence 4 (**Patched Phrase**). Here I've set the **TEMPO** to **100.00 BPM** and on the **Cowbell** track (**track 6**) I've assigned my version of the patched phrase cowbell loop (**old-cowbell-patched**) via pad **A05** of the **Cowbell Loops Demo** program.

Press **PLAY START** and the sequence plays with all tracks syncing perfectly at 100 BPM. Now, while the sequence is playing, increase the **tempo** to **120.0 BPM** – use the Q-LINKS if you prefer; the sequence tempo is assigned to **Q-LINK 13** (the **top Q-LINK in column 1**).



As the tempo changes, even the cowbell loop follows the tempo; in fact everything follows the tempo changes in real time. Try it at **70.0 BPM** – still works perfectly on all tracks.

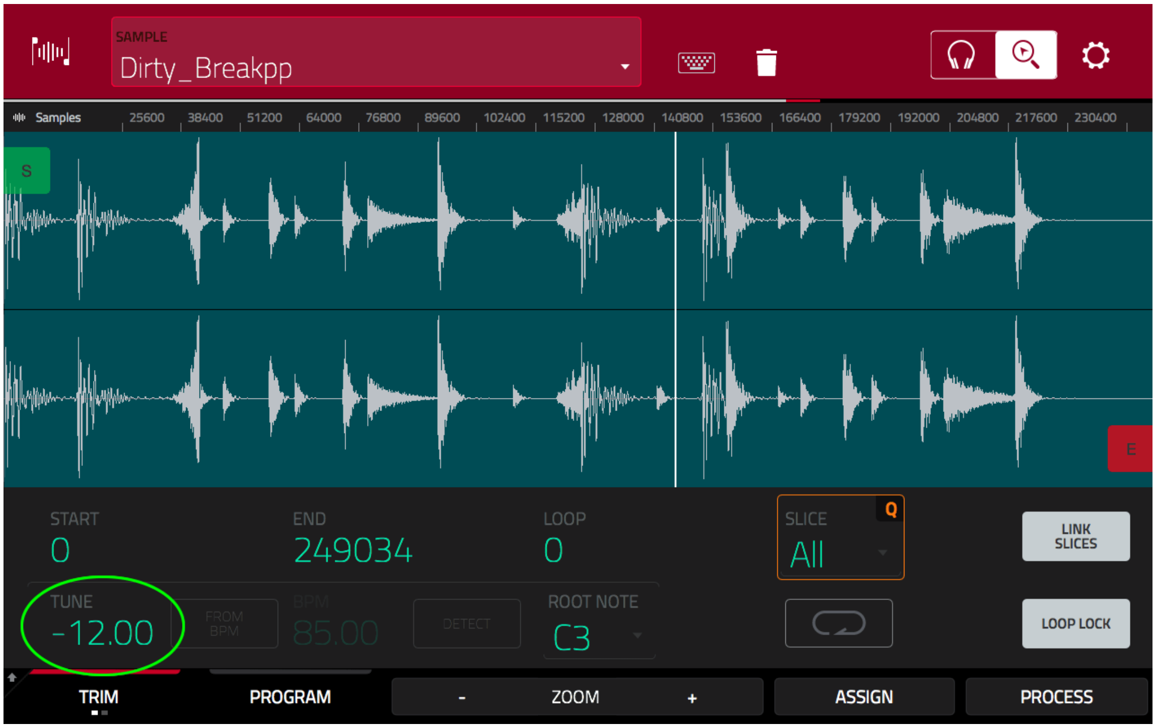
Apart from the tempo changing characteristics, a patched phrase loop is identical to any other looped sample. It's also a standard WAV file so can be played in any other audio application, although it will only tempo-match in the MPC Software.

The most important thing to remember when creating a patched phrase file is that you must first chop the break down to *individual hits*. If you leave any regions containing two hits then those two hits will always remain at the original tempo of the break as there is no way for the patched phrase file to realise that these are supposed to be two separately controllable hits – this will simply sound very odd when you begin speeding up or slowing down the loop.

It's also important that you clean up any end point clicks as these will be audible in the final patched break.

Tuning (Pitch Shifting) a Patched Phrase

Head back to **SAMPLE EDIT > TRIM**, select the **old-cowbell-patched** patched phrase sample and set a '**Tune**' value of **-12.00**.



Hit **PLAY START**. Your cowbell loop takes on a darker flavour – while the sequence plays you can continue to adjust the TUNE value in SAMPLE EDIT and your tuning changes will be reflected in real time. Try higher tunings to give the loop a very different feel.

Editing Patched Phrase Regions Before Conversion

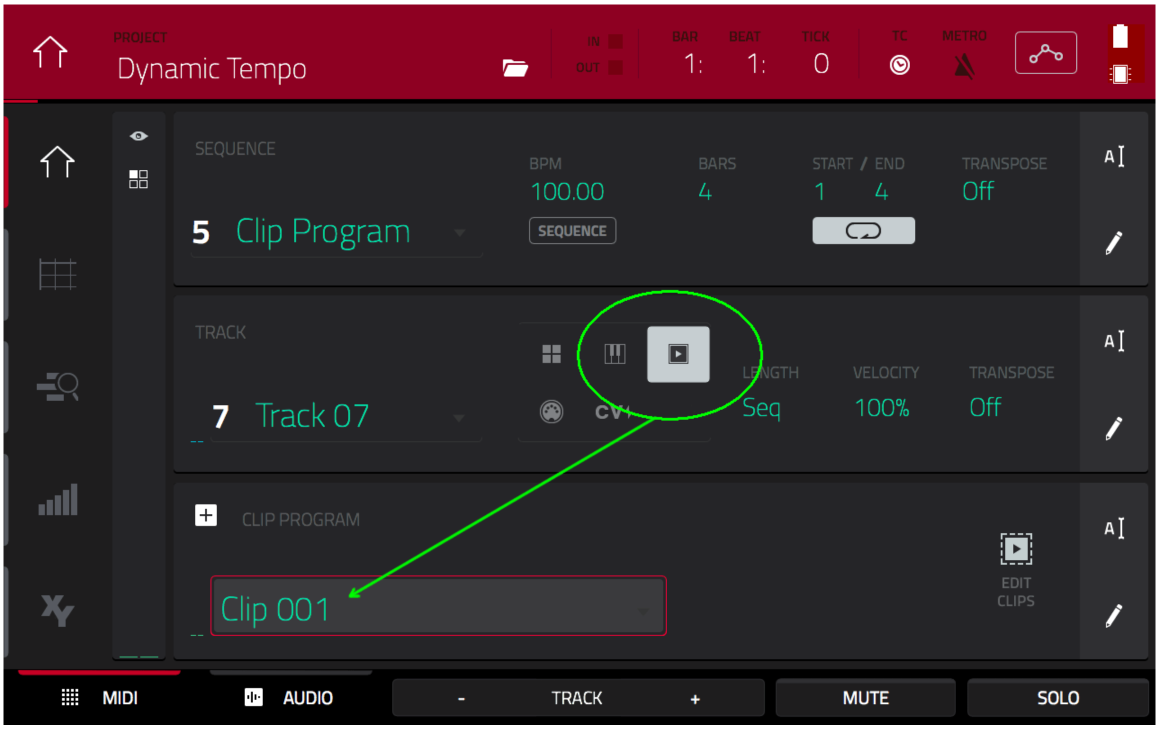
Remember a patched phrase starts out life as a standard chopped sample so each selected region can have any 'process' applied to it (**SHIFT > PROCESS**), such as reverse, silence, gain, fade in/out etc).

Warping in CLIP Programs

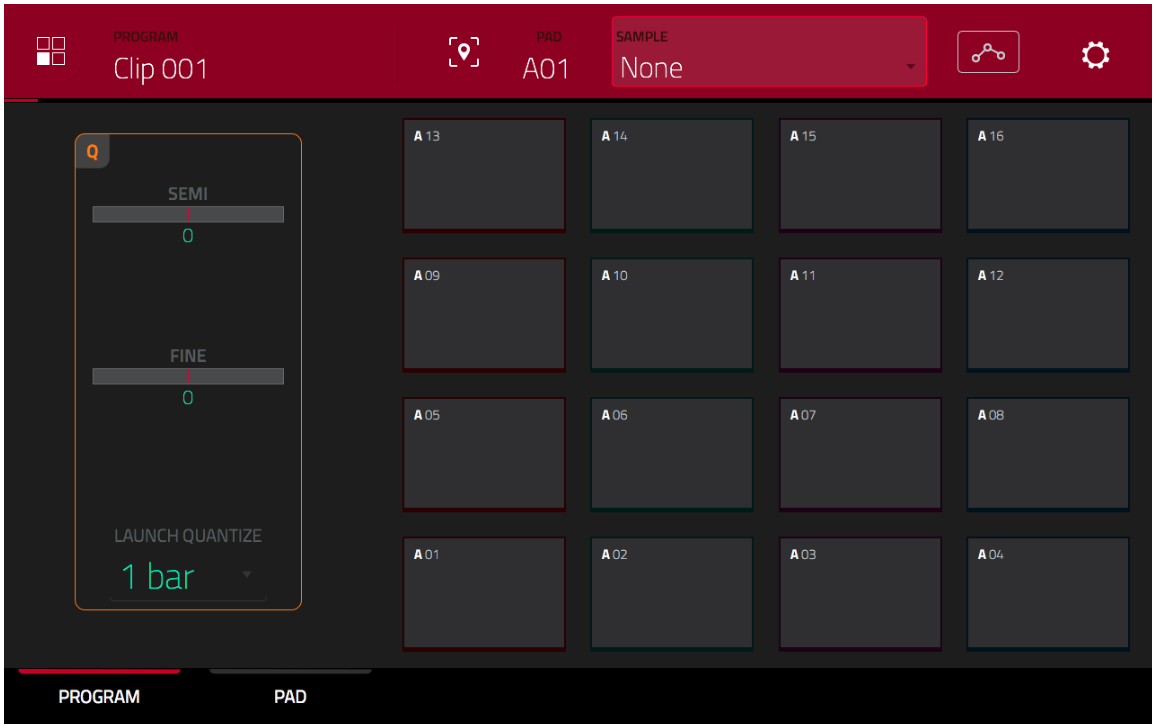
Warping is also included within **CLIP programs**. A CLIP program is a unique 'performance' program type optimised for launching loops and phrases in your beats. 'Clips' in an MPC are similar to clips you find in Ableton Live. A clip, or multiple clips, are 'armed' and placed in a queue before being automatically launched exactly on the next pre-defined 'quantise' cycle, ensuring their launch is perfectly timed with the currently played sequence.

While a clip program's main usage is typically in live performance where the producer can experiment in launching clips on top of his beats, there's no reason why you cannot use a CLIP program for normal 'pre-recorded' sequencing.

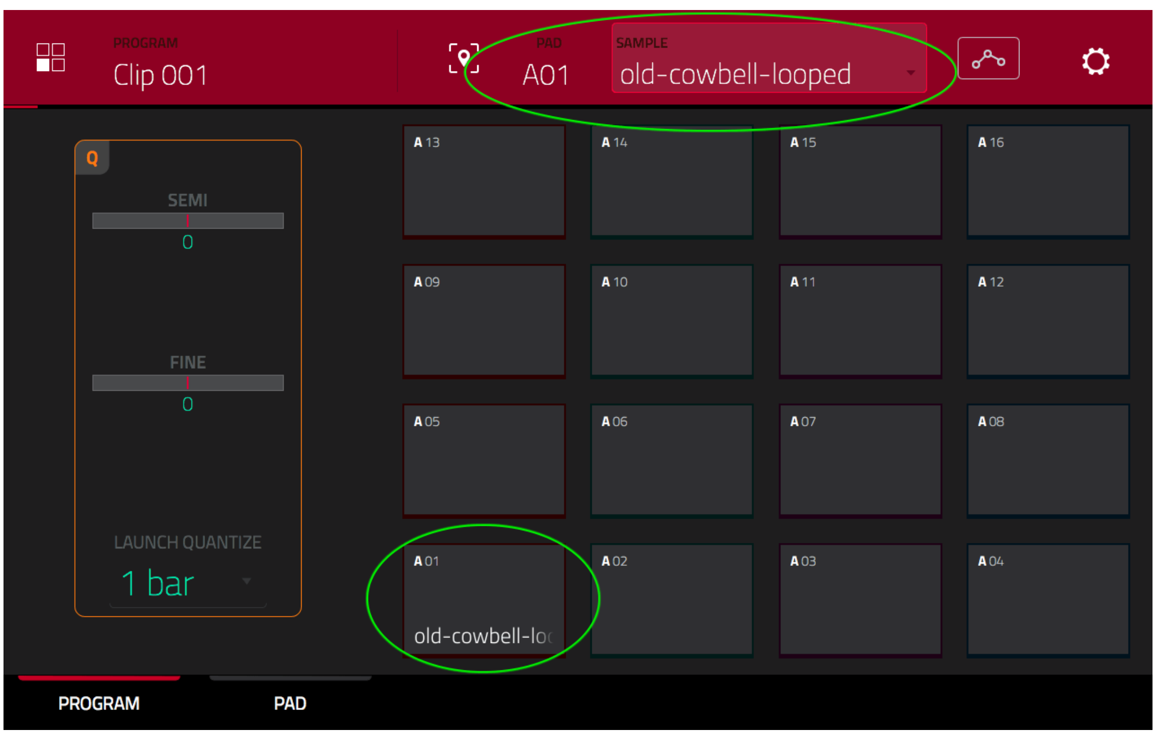
Go to **sequence 5** (**Clip Program**). This is our usual Demo Beat, this time at 100BPM with no cowbell loop. Go to the empty **track 7** and on the **TRACK** row tap on the **CLIP icon**:



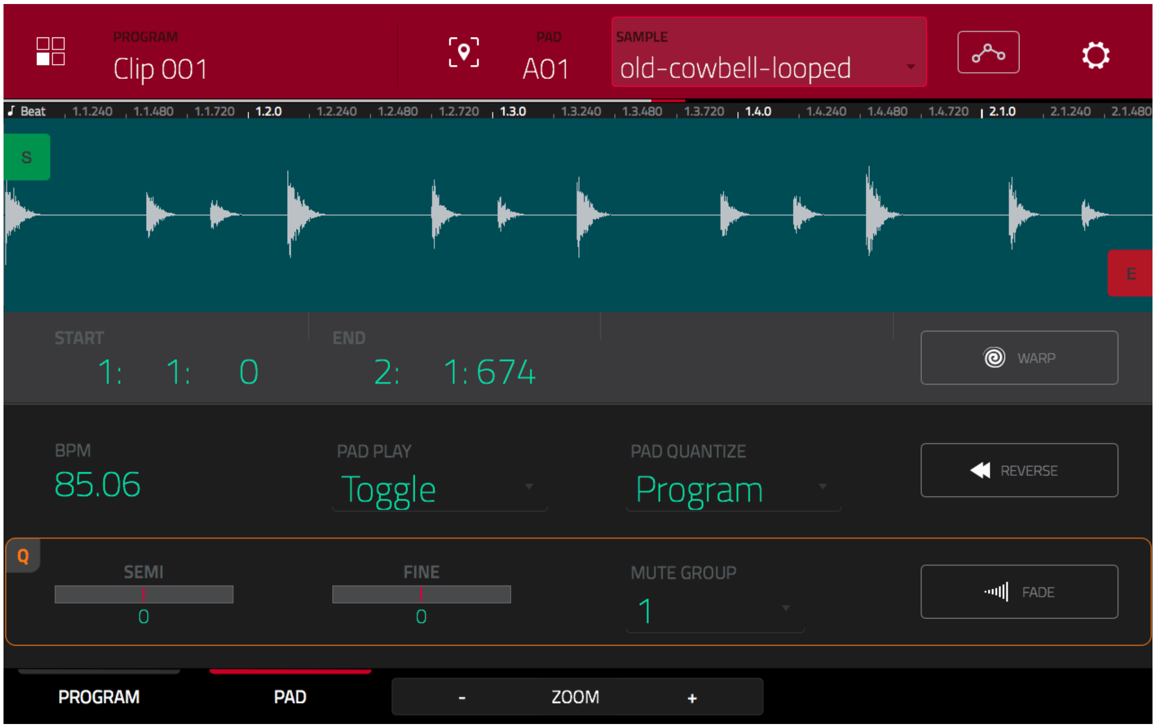
This creates a '**CLIP'** program track – assign the blank **Clip 001** program to this track (don't use the 'Clip Demo' program). Go to **PROGRAM EDIT** (or hit the **EDIT CLIPS** icon on the bottom right of the MAIN screen).



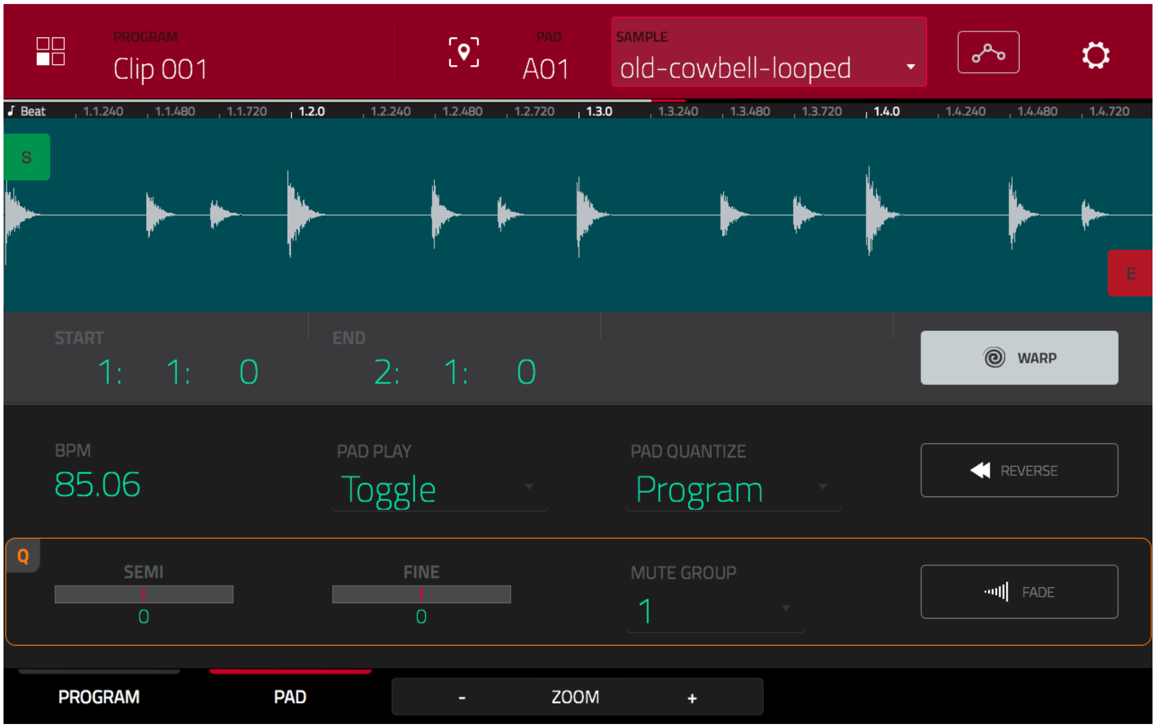
In the main '**PROGRAM**' screen we can begin assigning samples to our clip program. Tap on pad **A01** to choose A01 as the target pad (the pad doesn't light up, but it shows A01 at the top of the screen). At the top right of the screen tap on **SAMPLE: None** and change to **SAMPLE: old-cowbell-looped**.



Tap on the **PAD** button at the bottom of the screen to go to the **PAD** sub screen



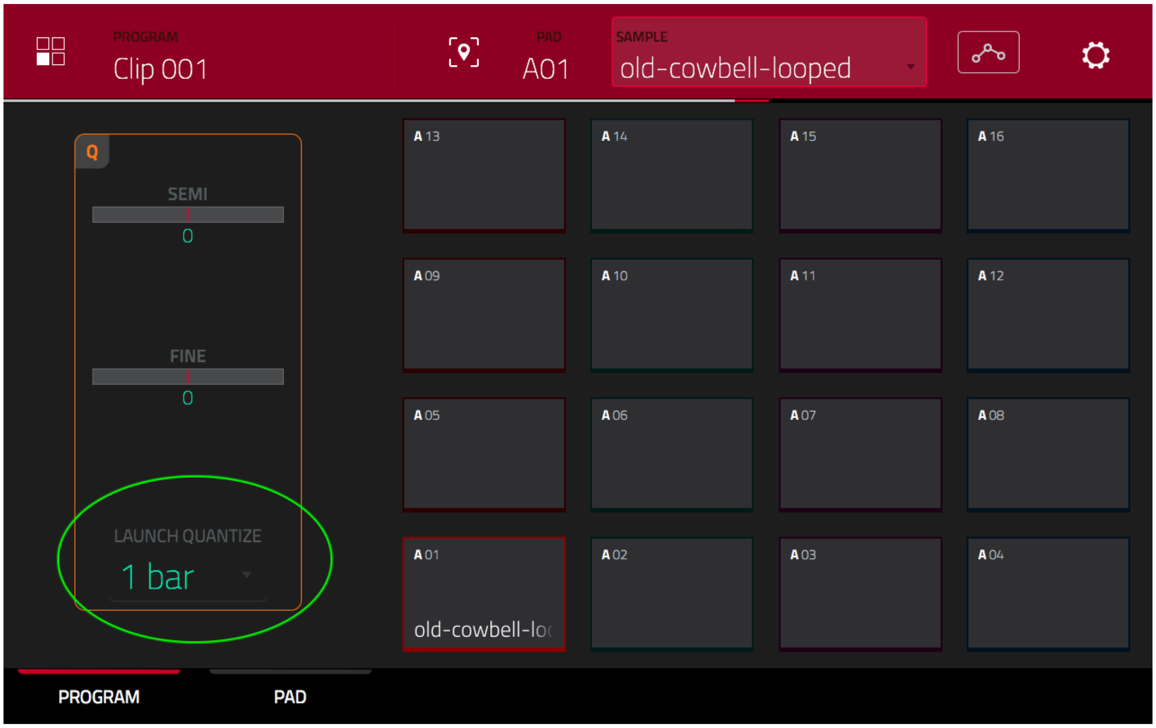
There are a few parameters we can play with here, but for the purposes of this part of the tutorial, the main parameters to focus on are the BPM setting and the WARP button. As you can see, the CLIP program recognises that the cowbell loop has a tempo of 85.06 BPM, which isn't going to match our current sequence tempo of 100 BPM. So, hit that **WARP** button.



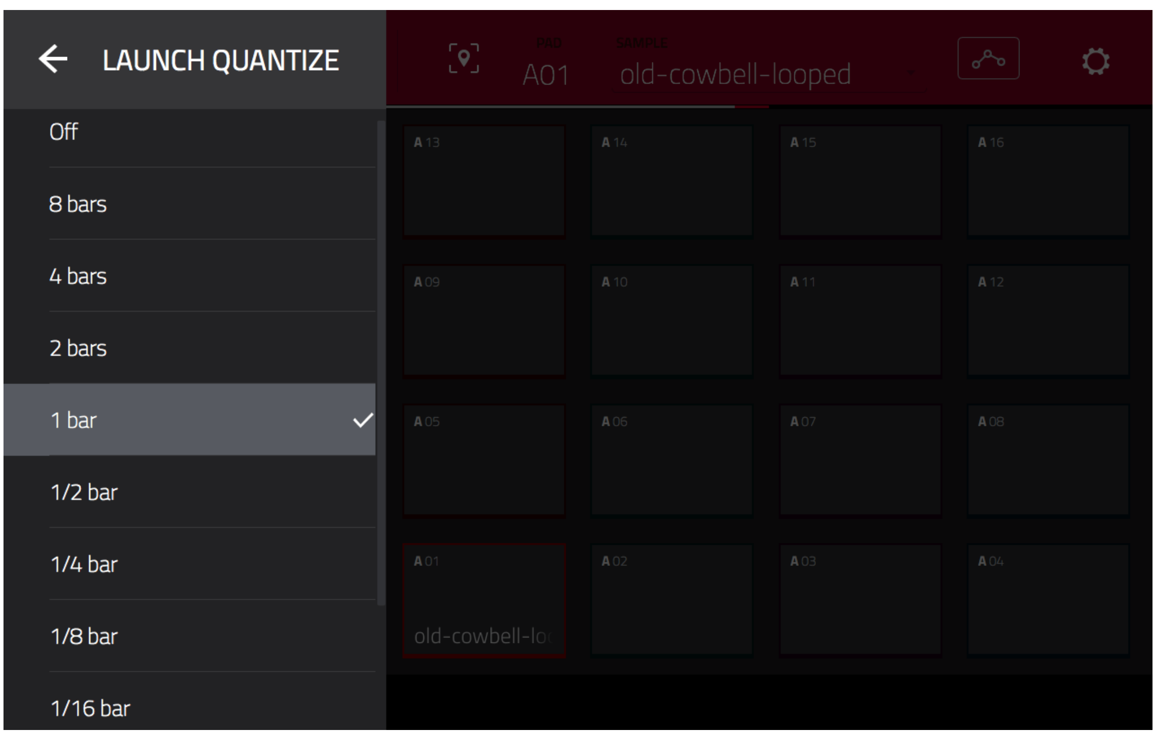
Press **PLAY START** so the sequence begins play back. While it is playing, hit pad A01 – try to trigger it at an odd time, not at the start of a bar.

You should notice that when you hit the pad, the cowbell loop is not triggered immediately like it would in a drum program. Instead, the loop only begins playback at the start of the *next bar*. It's also 'warping' perfectly in time with the 100 BPM sequence.

When you hit a pad in a clip program, the sound is effectively added to a queue and is not launched until the next PAD QUANTIZE point is reached. While it is in the queue you'll see the pad flash yellow. Refer back to the **PAD** screen and you'll see that **PAD QUANTIZE** is set to **Program**. This simply means that this pad obeys the 'master' **LAUNCH QUANTIZE** setting for the program itself. To see this, go back to the **PROGRAM** screen:



Here you can see the **LAUNCH QUANTIZE** is set to **1 Bar**. This means that each time a pad from this is hit, the sample is queued and isn't played until the start of the next bar is reached. Double tap **LAUNCH QUANTIZE** and you'll see there's plenty of different quantise options to choose from.



If you want to queue all pads to only come in at the beginning of this sequence set LAUNCH QUANTIZE to 4 Bars. If you want to disable this behaviour, set LAUNCH QUANTIZE to Off and pad triggering will mimic that of a DRUM program (i.e. instant trigger).

If you want certain pads to have their own unique launch quantize, just go back to the PAD screen for those pads and set PAD QUANTIZE from 'Program' to whatever quantize you wish.

Just leave the **LAUNCH QUANTIZE** set at **1 Bar** and return to the **PAD** screen. The other setting of relevance here is **PAD PLAY**; this can be set to **Toggle** or **One Shot**.

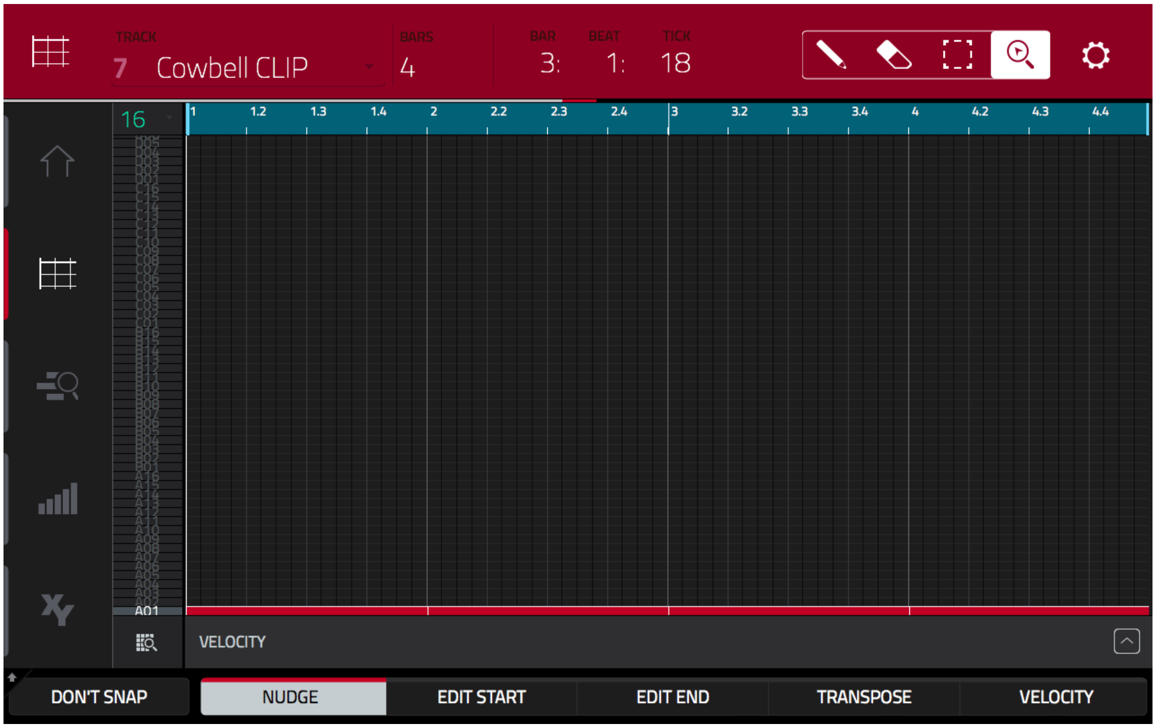
With the default **'Toggle'** you'll notice that once the cowbell is triggered it continues to play in a loop for as long as the sequence plays. To turn off the loop, press the pad again and the loop will stop *at the next quantize point*; in the case of our current '1 Bar' setting, this would be the end of the current bar.

If you set **PAD PLAY** to **ONE SHOT** the loop will just play through once, just as if it were in a DRUM program assigned to a ONE SHOT pad.

Leave **PAD PLAY** set to **Toggle**.

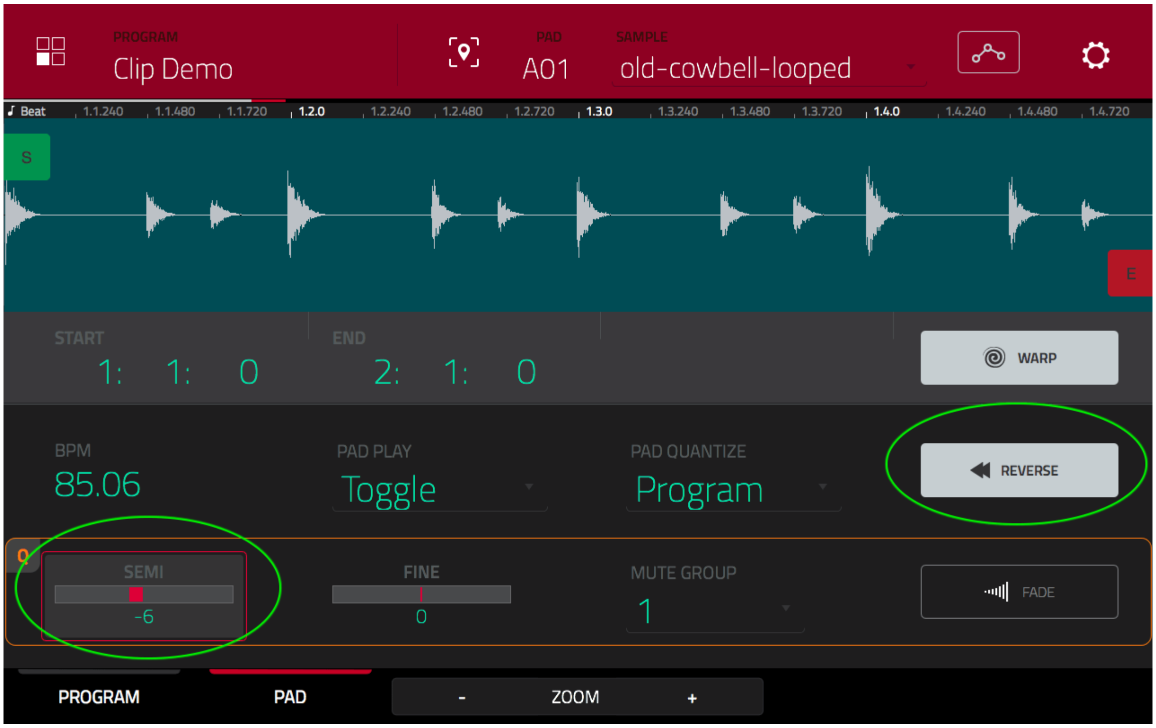
To record a continuous cowbell loop across our entire sequence, simply hit **REC** and **PLAY START** and during the count in period, hit pad **A01** and it will enter the 'trigger queue' – as soon as the sequence begins recording, the loop is automatically triggered and the events recorded to the sequencer track.

After recording the 4 bars, press **STOP** and take a look at the events in **GRID EDIT**.



As you can see, the sequencer just records the events as if you had been holding down pad A01 continually for each individual bar. These events are however no different to any other events, so you can if you wish edit those events; for example, you could change the duration, delete, move etc.

Select **sequence 6** where I've mimicked the note durations that we used in previous examples, with the pauses in the loop in bars 2 and 4 – I did this simply by selecting those events and using **EDIT END** in **GRID EDIT**. I've also hit the **REVERSE** button and dropped the **SEMI** parameter down to **-6** in the **PAD** screen for this clip.



Don't forget that the warping pitch control and reverse play options are also available in DRUM programs, so you can use which ever 'interface' suits your workflow.

019 Non Destructive Chopping Techniques

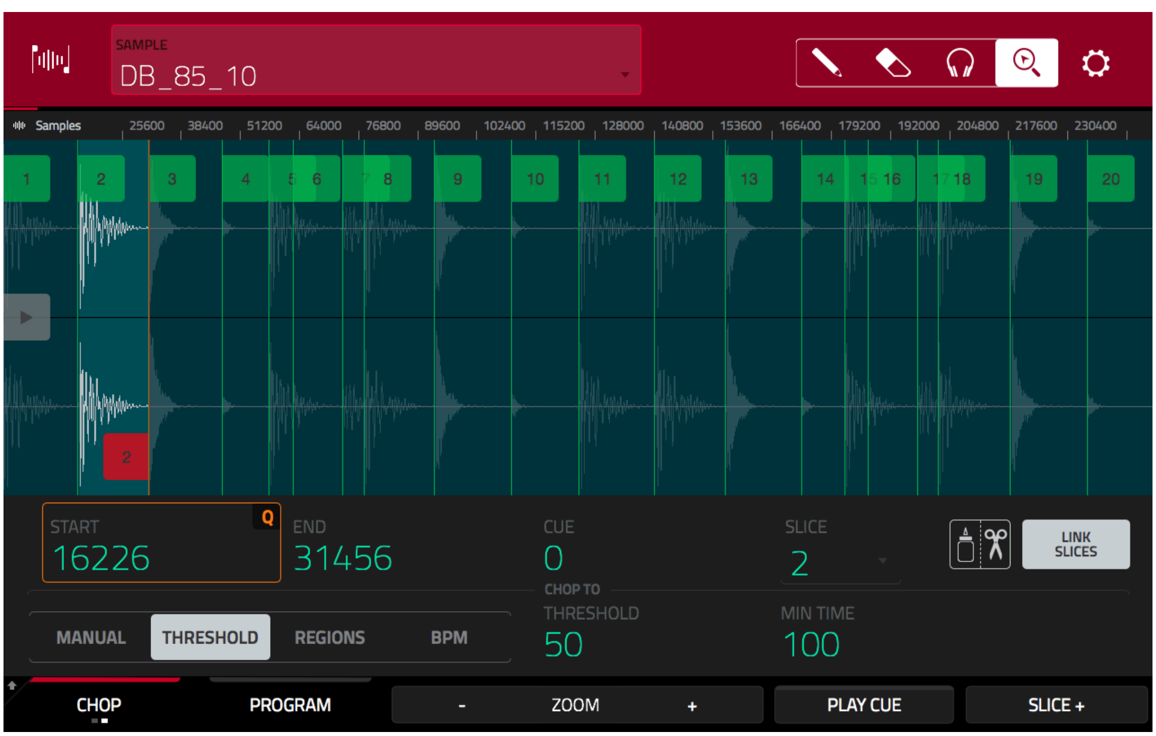
In this chapter we'll be using an advanced MPC chopping feature called 'Non Destructive Chopping' (NDC), which opens up a whole world of creative chopping ideas!

Introduction to Non Destructive Chopping

To help explain the concepts behind Non Destructive Chopping (which I'll just refer to as '**NDC**' from now on), let's initially move back to familiar territory. Firstly, start a new blank project.

From the **chapter 19** folder load up the sample **DD_85_10**, which is a short drum break from Timmy Rickard's 'Dirty Drum Break' collection (http://www.mpc-samples.com/product.php/207/dirty-drum-breaks/).

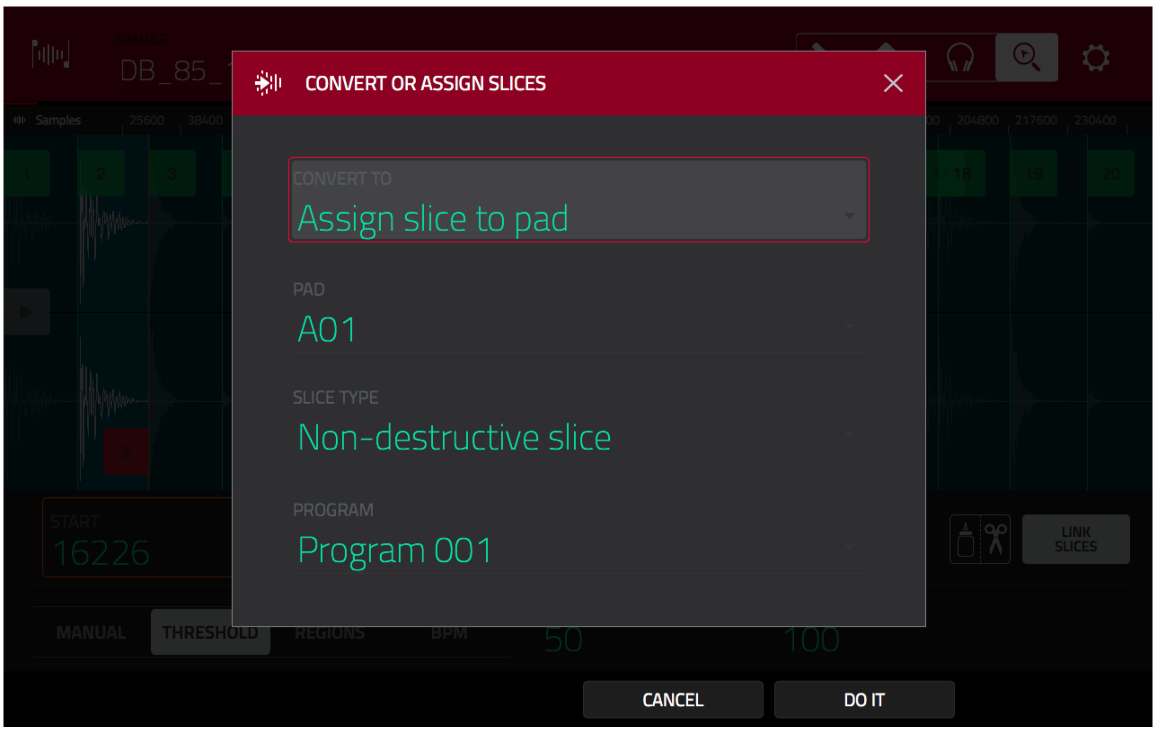
As we did in previous chopping examples, head over to **SAMPLE EDIT** > **CHOP** > '**THRESHOLD**' and increase the 'Threshold' value until you can see that all individual hits have been isolated as unique slices. A **threshold** of **50** should do nicely:



So far, everything is as you are already used to. We're now going to build a drum kit from this break, but using non-destructive techniques and as such we can leave any 'fine' region edits until later.

Let's quickly build a drum kit with our best slices. First I want a nice kick – preview all the regions and you'll hear that there's a lot of standalone kicks in this break; slices 1, 2, 5, 7, 8, 9, 10, 11 and 18; let's select **slice 2** as that's a nice clean one with no hat on top, so press pad **A02** to select this.

Select '**CONVERT TO: Assign slice to pad**':



With '**CONVERT TO**' set to '**Assign Slice to Pad**', this conversion process is going to place the *currently selected slice* on to any pad we choose in the currently selected program

Hit pad **A01** so our kick slice will be assigned to pad A01.

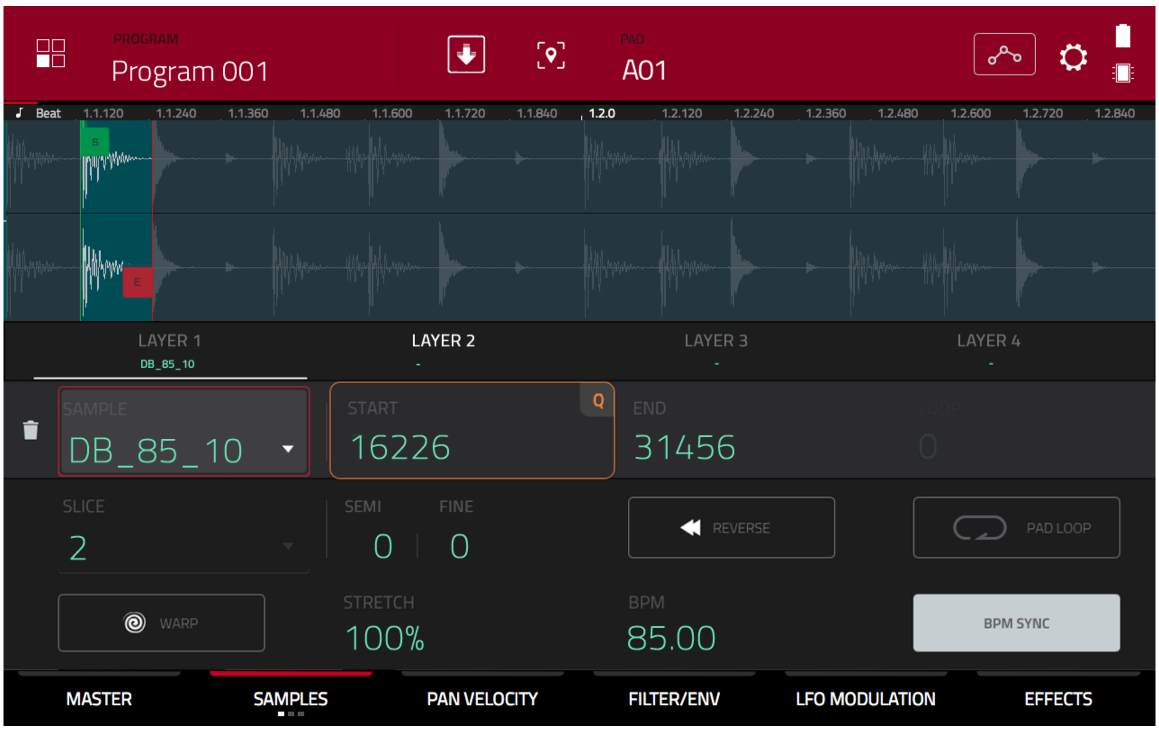
'**SLICE TYPE**' should be set to '**Non-destructive slice**'. And under '**PROGRAM** you can select which program the slice should be sent to. As this is a new project, this slice will be sent to the default '**Program 001**' blank DRUM program.

Hit **DO IT**. This kick slice (region 2) is now assigned to pad **A01** in our program – at this point you can quickly check this by going to MAIN or Prog Edit and press pad A01 to hear the kick. However, let's stay in CHOP for the moment and assign a snare and hat to this program first.

So in **CHOP**, locate a good snare slice – let's use **slice 9**. Press **pad A09** to select this slice and once again press **SHIFT** and **CONVERT** to bring up the '**Convert or Assign Slices**' screen. This time assign our slice to pad **A02**, so hit pad A02 and then hit **DO IT**.

Finally, locate a nice hi hat slice – try **slice 4**. So select **slice 4**, go to **SHIFT** and **CONVERT** and assign this slice to pad **A03**. Hit **DO IT**.

So with our kick, snare and hat assigned to pads A1, A2 and A3 respectively, let's check out our new program in more detail. Hit pad **A01** and go to **PROGRAM EDIT > SAMPLES:**



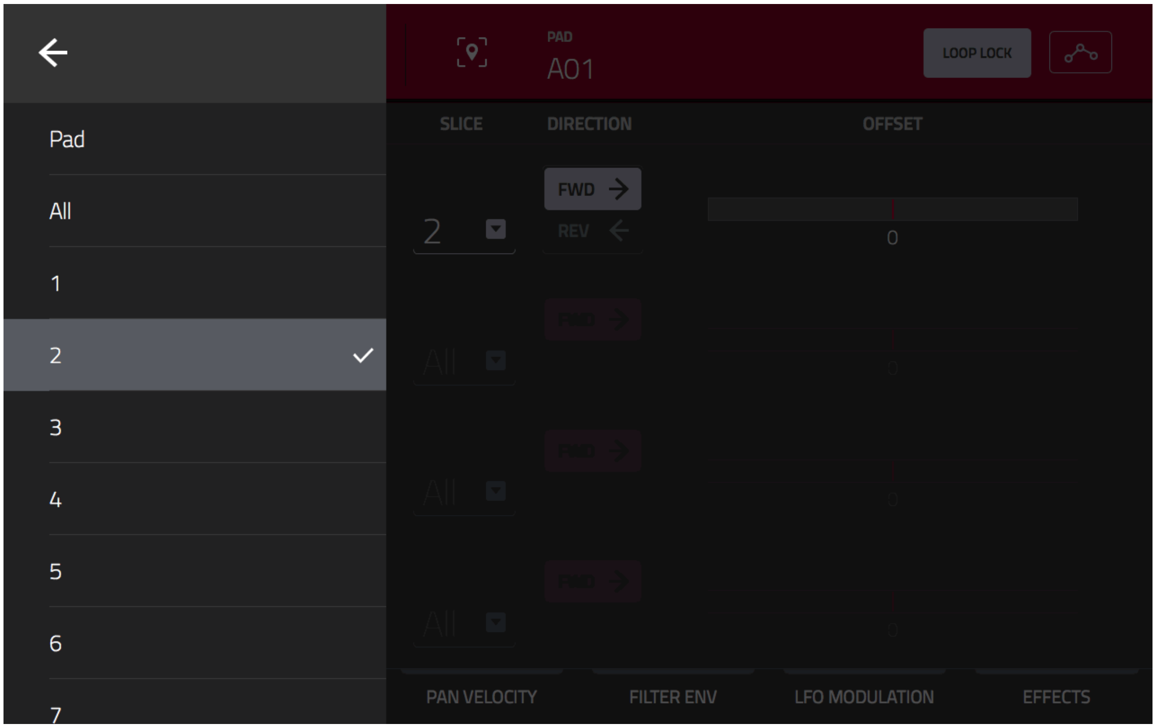
As you can see, rather than having a new 'chop' sample assigned to **LAYER 1** of pad **A01** we instead have the original **DB_85_10** break assigned to it. However when you strike it, only the kick of slice 2 is played back, not the entire break.

The fact that only slice 2 is playing back is shown in two areas on the SAMPLES page. Firstly you can see the entire break is displayed in the waveform, however only 'slice 2' is highlighted, the rest of the break appears 'inactive'.

Secondly, the **SLICE** parameter now reads '**2**' instead of '**All**'.

When we assign a sample to a pad layer that has been sliced in the CHOP, the SLICE parameter, as the name suggests, allows us to assign any specific **slice**

region to that layer. In this case we assigned slice 2. But double tap on the drop down and you'll see you can assign *any* slice from our chopped break.



And setting the slice to '**All'** will set the pad layer to the default setting of playing the *entire* break (we'll look at the 'Pad' option later).

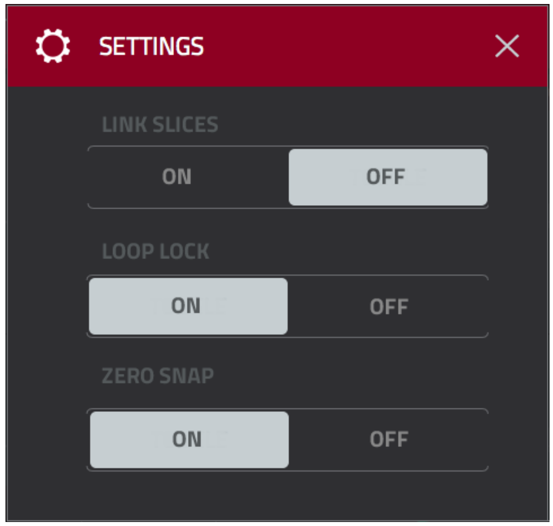
Hit pad **A02**, to which we assigned the snare (**slice 9**). As expected, this pad layer has a **SLICE** value of **9**. And the hi hat on pad **A03**, which uses **slice 4** from the break, has a **SLICE** value of **4** set.

One key point to realise is that, unlike destructive chopping, we have not created any new samples here - no copies, no actual 'extractions'. We still only have a single sample stored in the project, **DB_85_10**. All we are going do is selectively accessing the slice regions of our break that we set in the CHOP screen.

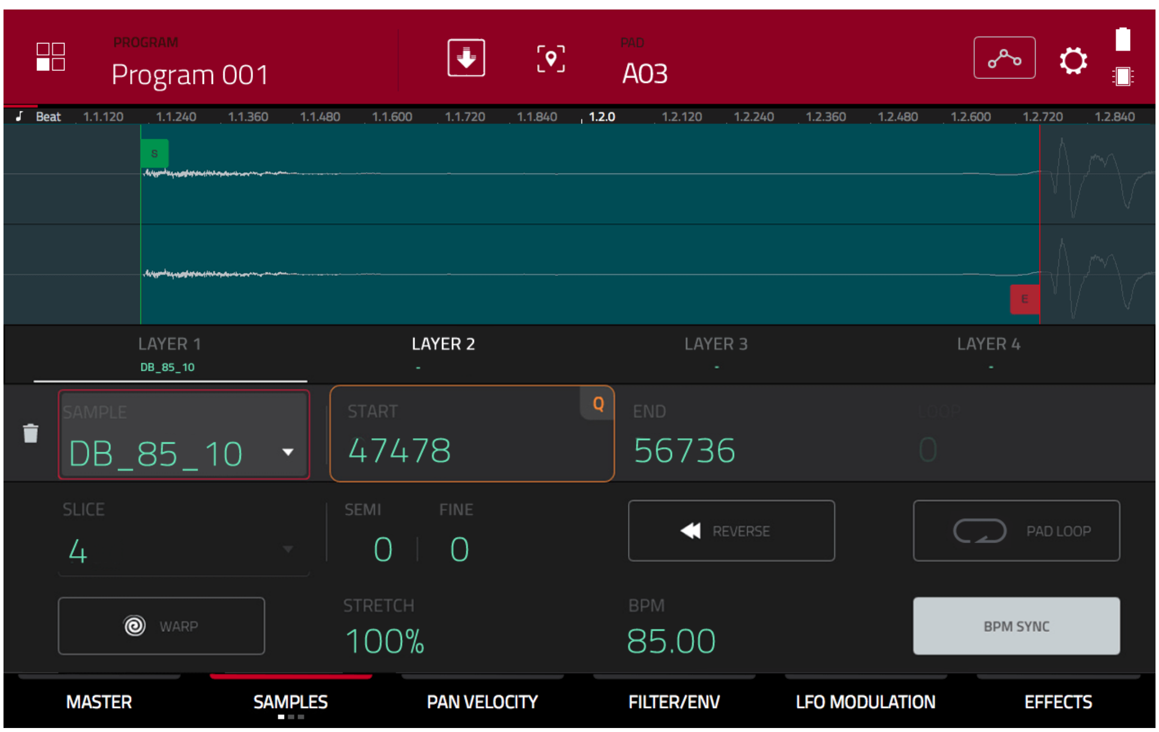
Avoiding the need for dozens of samples is obviously a very efficient way to work and it's still surprisingly flexible as the MPC program itself still treats these slices as if they were standalone samples, so all program parameters, filters, envelopes and LFOs can be applied just as you would with any other type of sample.

The waveform displayed in the SAMPLES page is not just for visual effect, we can use it to make adjustments to the START and END points of our slice. Select pad **A3** to hear our hi hat **slice 4**. As you can hear the end of this slice features a brief click, so let's make a quick adjustment to the end point of this slice.

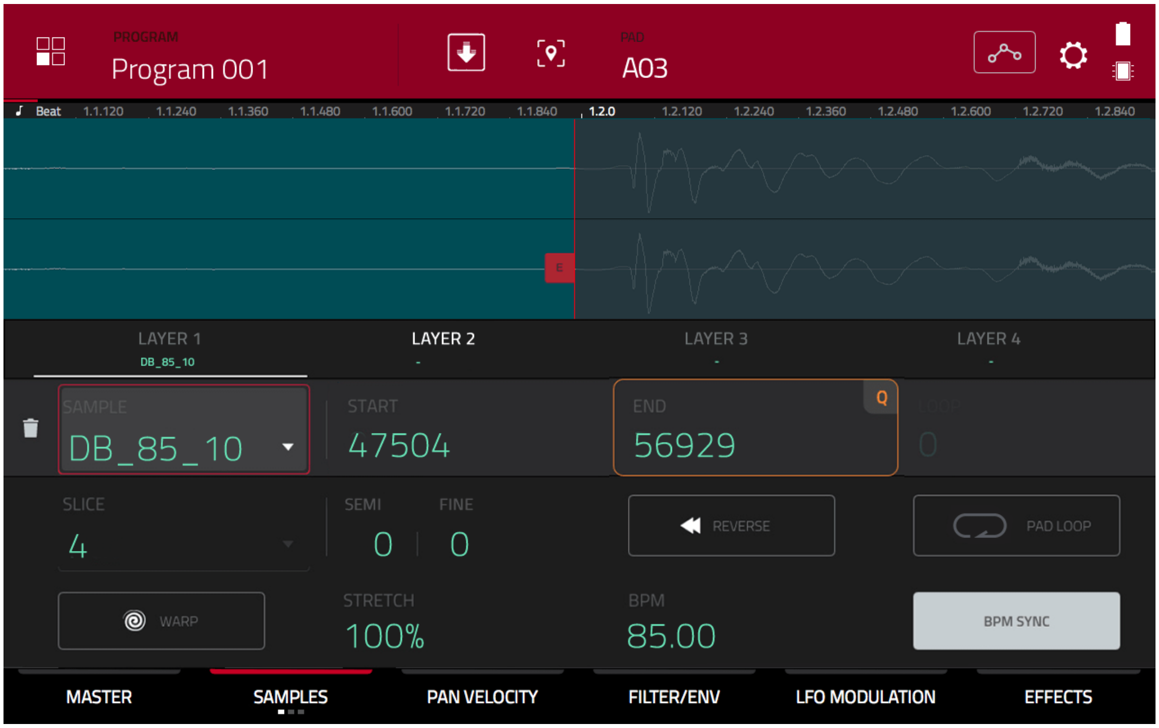
Firstly, it's important to turn set **LINK SLICES: OFF** – this way any changes we make to the edit points of the slice will have no impact on any adjacent slices. To do this, hit the **SETTINGS gear icon** in the right of the screen:



While you are there, make sure **ZERO SNAP** is '**ON**' as well. Now pinch and zoom in to attain a good magnification:



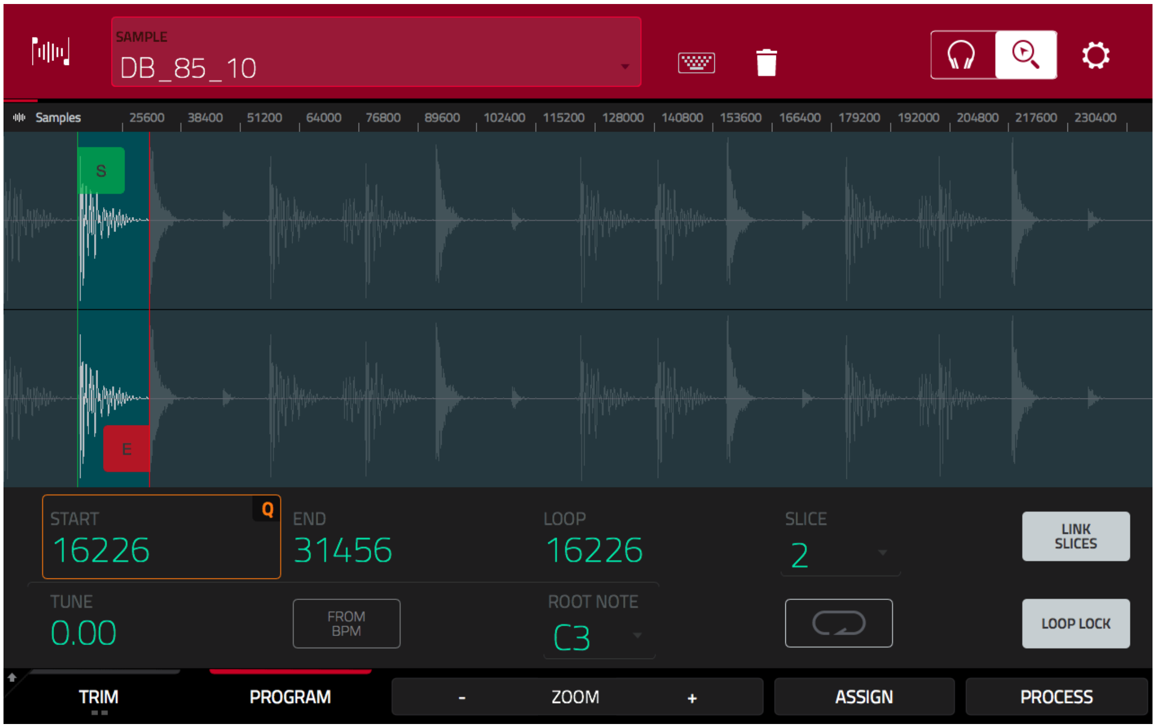
Now adjust the end point of the slice to remove the 'click'. You can use all the usual editing options - for example, tap on the END point and turn the data wheel anticlockwise, or tap on the red END point flag on screen and drag it to the left, or use the **bottom Q-LINK column 2**. Set an end point around **56929**.



Preview our slice using pad **A3**. Perfect.

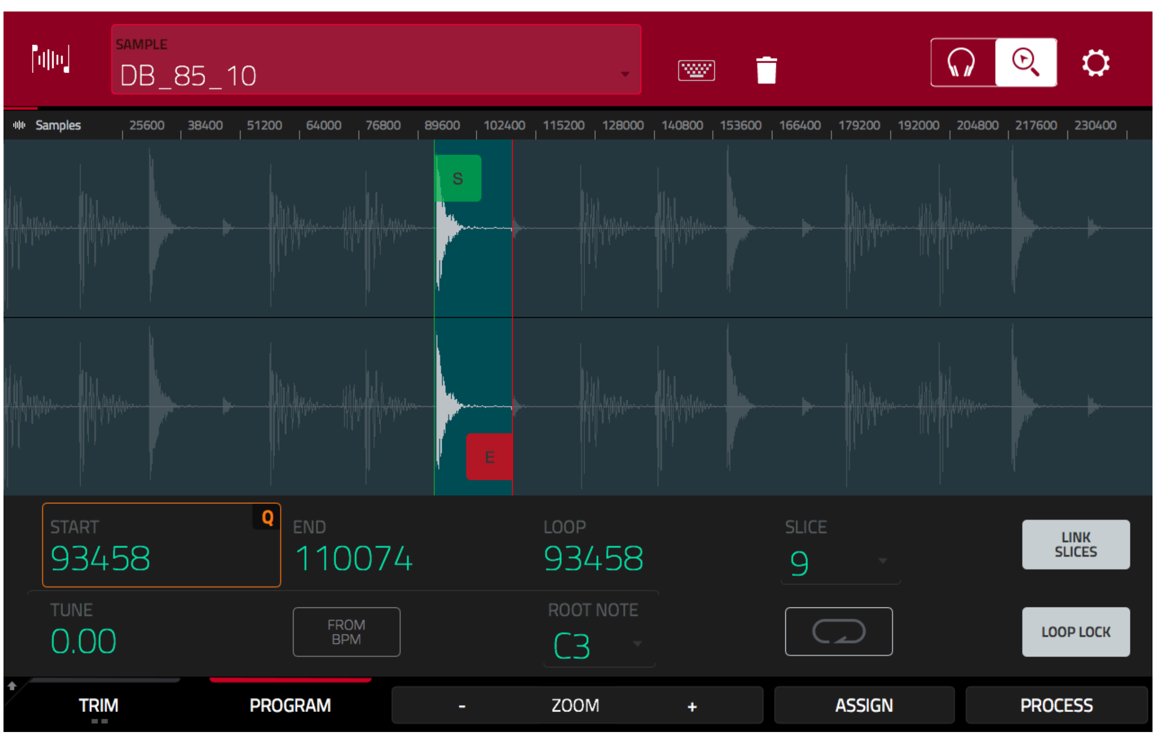
PROGRAM Chop Mode

Head back to **SAMPLE EDIT** > **CHOP**, where you'll see our sliced break beat. To assist us in working with a program that utilises slices, the MPC features a third SAMPLE EDIT view; hit the dedicated **PROGRAM** button:



In **PROGRAM** view, we now see our sliced break in relation to the way its slices are assigned to the current program. Hit pad **A01** and the waveform actually highlights **slice 2**, indicating that the slice 2 region currently assigned to pad A01.

Now hit pad **A02** and you'll see that **slice 9** is assigned:



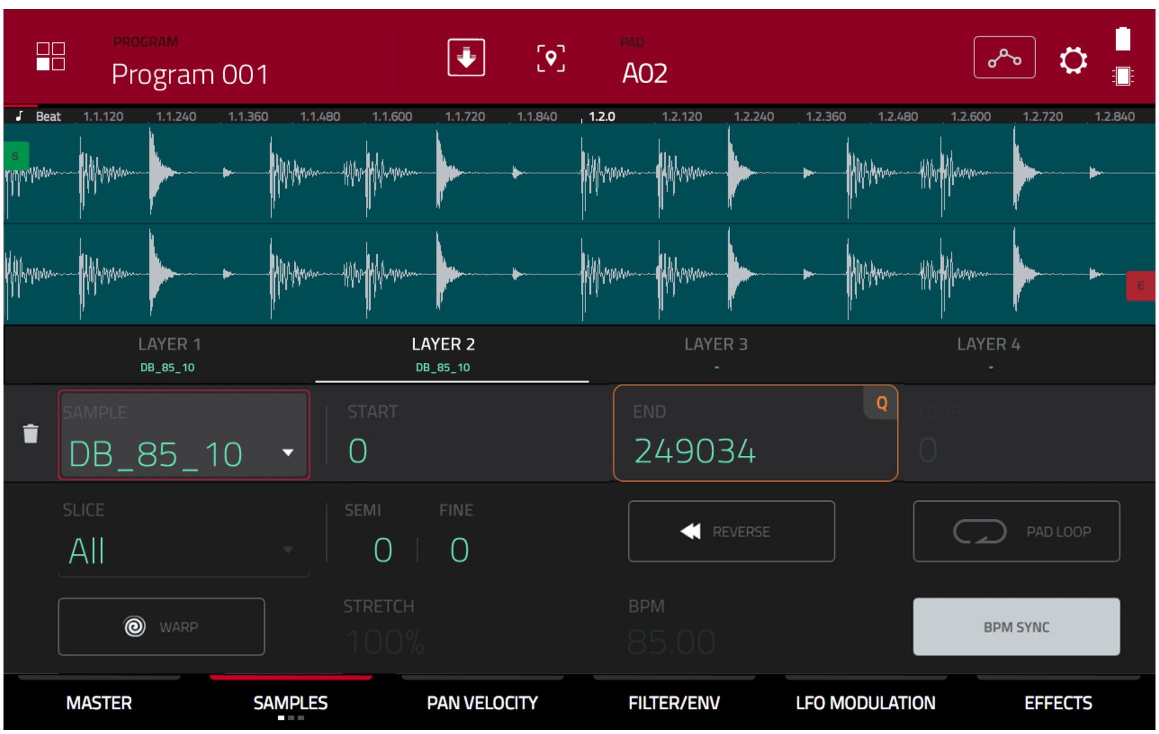
This is identical to the way your slices are displayed back in the SAMPLES page. One advantage of editing in SAMPLES EDIT > PROGRAM is that you can apply the sample PROCESS option to each slice; PROCESS is not currently available in PROGRAM EDIT.

However, one disadvantage of this PROGRAM mode is that there is currently no way to select which pad *layer* you wish to edit, and this is going to be important in the next example.

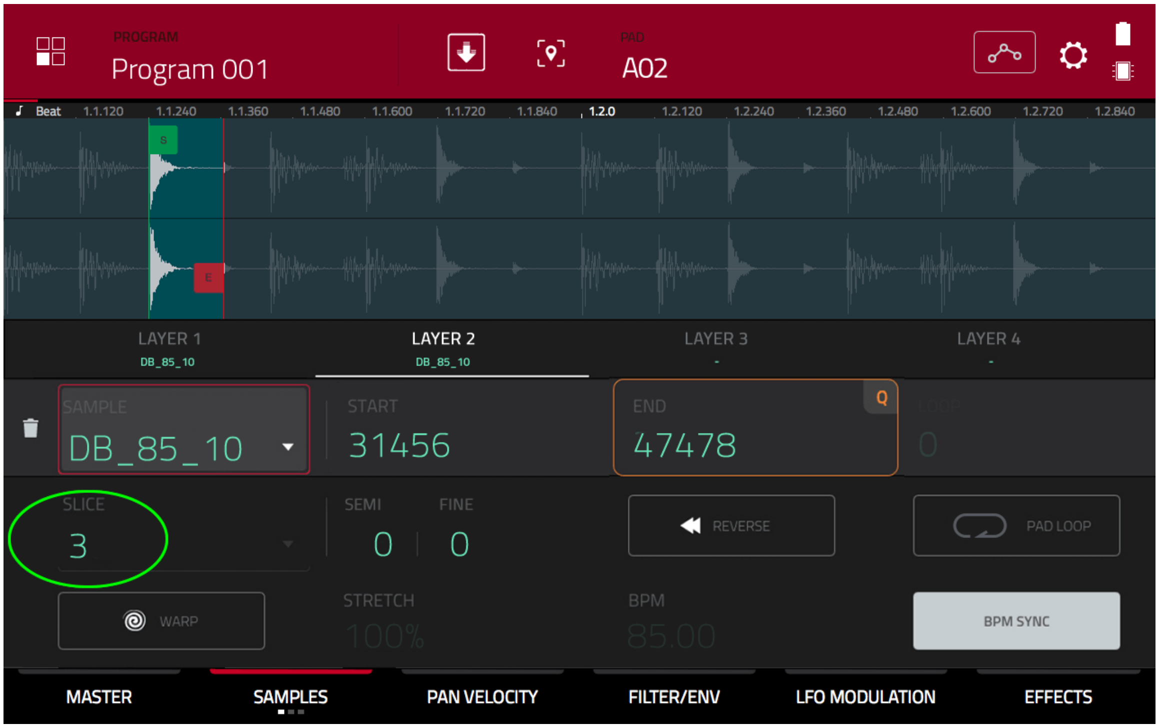
NDC Over Multiple Layers

Previously we used slice 9 for our snare, but in our break we have other snares we can use; run through our slices and find all the possible snares - slice 3, slice 13, and slice 19. Now we know from chapter 17 that we can set up a pad to play random or cycled layers, so let's set up our snare for more realistic playback using our NDC slices.

Head over to **PROGRAM EDIT > SAMPLES** and select pad **A02** (our snare pad). Tap the **LAYER 2** tab and assign the **DB_85_10** sample to this layer:



At this point if you hit pad **A02**, the snare slice on layer 1 plays, but the entire break also plays on layer 2. This is because the **SLICE** parameter is currently set to **ALL**. Change this so that the **SLICE** parameter plays slice **3**.



Now go to **LFO MODULATION** and change the **LAYER** setting from VEL to **CYC**. Press pad **A02** several times to hear it switch between the two different snare samples.

You can take this further by adding snares to the other two layers, using **slices 13** and **19**. Then move over to the kick on **A01** and repeat this over all four layers using **slices 2, 8, 11, 18. F**inally for the hat on pad **A03**, use **slices 4, 10, 14, 20**. Remember to set **LAYER** to either **CYC** or **RND** for all three pads.

Editing Layers

Some of our layers unfortunately contain 'clicks' at the end of the slice and you'll soon discover that while the SAMPLES page gives you quick access to edit each layer's slices, there's no easy way to solo the audio for one particular layer so you can hear it *in isolation* while you make any changes to the start and end points.

For example, keep hitting pad **A01** to hear the four kicks cycle through and you'l soon hear that two of the kicks have clicks on the end; but which ones?

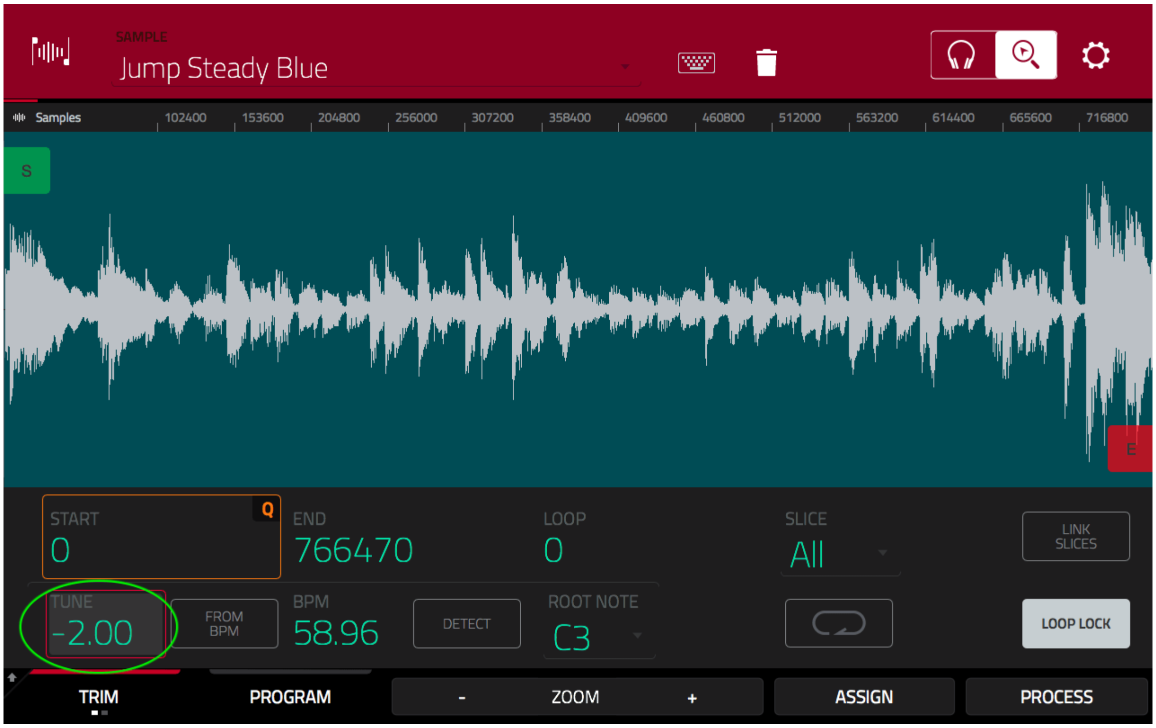
Some kind of layer 'solo' option would be nice and hopefully Akai look into implementing this in a future update, but in the interim I feel the best option is to just head back over to SAMPLE EDIT > CHOP and simply check all your slices and remove at end point clicks using the techniques I have discussed previously in the book; any tweaks you make are instantly reflected in your program.

You can load my version of this kit from the **chapter 019 folder**, **NDC Slice Kit.xpm**. Notice that if you select the **DD_85_10** sample in CHOP, the custom slices you created have been saved within the sample itself.

So we now have a responsive drum kit that features twelve unique sounds, but only uses a single sample to power it. And at any time all twelve sounds can be easily tweaked and edited simply by editing their corresponding slices.

Chopping Musical Pieces

Load up the sample **Jump Steady Blue** and select it in **SAMPLE EDIT > TRIM** – it's an excerpt from an old 12-bar blues tune. This is not sequenced at all and is played quite freely so it is going to be harder to chop compared to a more tightly played drum break. In the next example I want to use this blues loop to create a beat with a laid back, hip hop feel. Firstly, let's give it a darker vibe simply by tuning the entire sample down. As we've seen previously, this can be quickly achieved using the '**TUNE**' parameter in the **TRIM** screen. Let's retune this sample to **-2.00**.



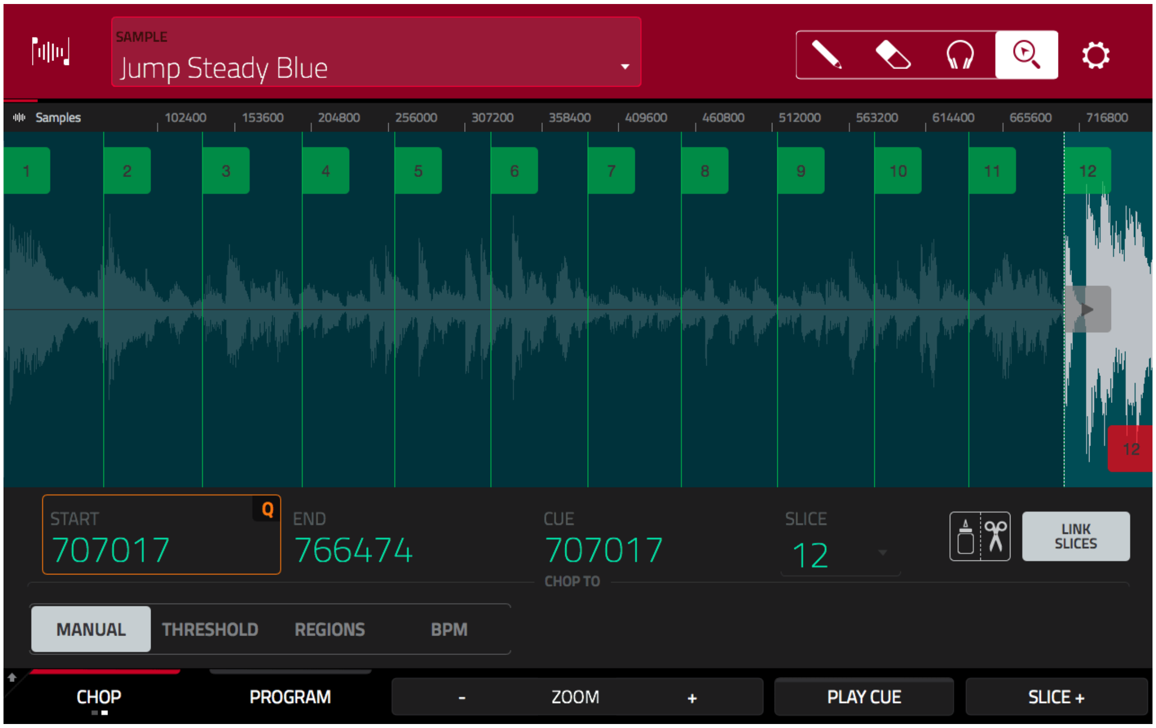
Select the **CHOP** screen, make sure '**LINK SLICES**' is **ON** and **Pad Audition** is set to **ONE SHOT** – to do this, hold down **SHIFT** and press **ONE SHOT** so it becomes a grey/red tab.

Musical samples like this are usually impossible to chop with 'threshold' chopping due to the lack of definition in the waveform peaks. Rather than 'auto chop' we are going to simply play back the entire sample in CHOP mode and dynamically add slice points 'by ear'.

It's going to be tricky to add our first slice point in real time, as this needs to be placed at the very start of the sample, so instead take the **CUE** parameter to **0** and now manually add a marker here by tapping **pad 1** – not only will this add the first slice marker at '0', it will also begin playback of the sample.

Try to get a feel for where you want to insert your pad slices. You can preview the sample for as many times as you wish until you are ready to start manually adding your slices - let's try to add a slice point at each bar of music.

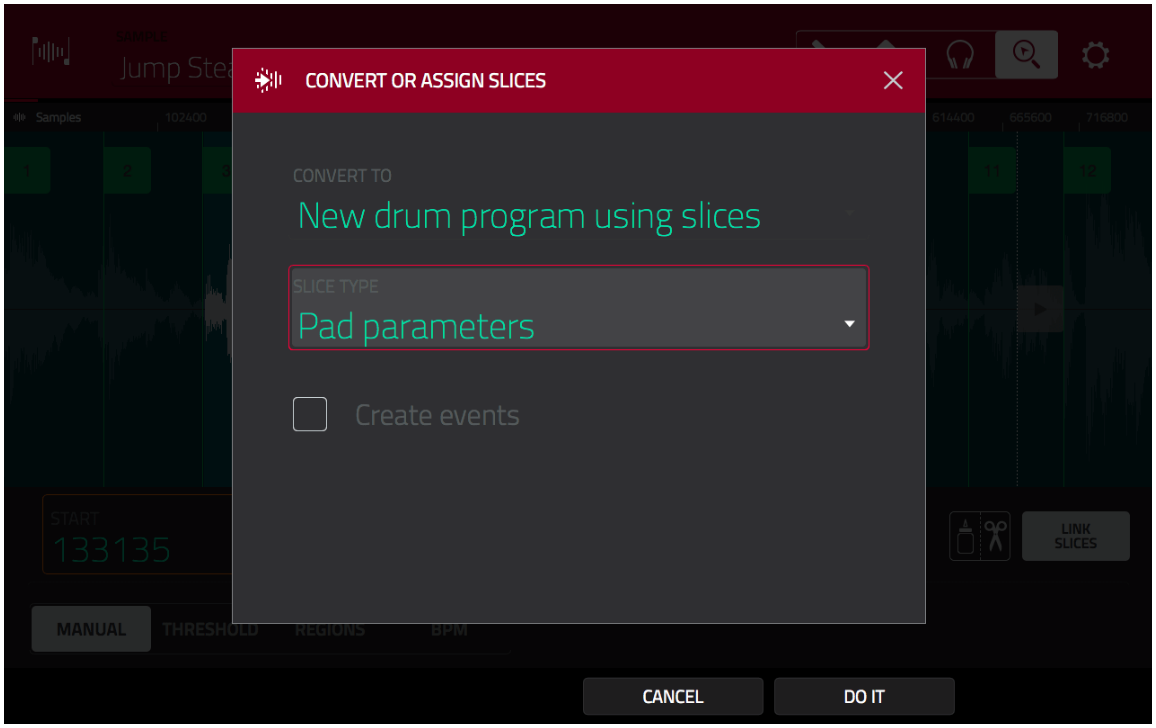
So, hit pad 1 to begin playback again. At this point, you can now press the on screen **SLICE+** button whenever you wish to add a marker. Here's what I came up with:



If you want to remove a slice, hit the relevant pad to select it, hold down **SHIFT** and hit **SLICE-**. To remove all slices hold down **SHIFT > CLEAR ALL**.

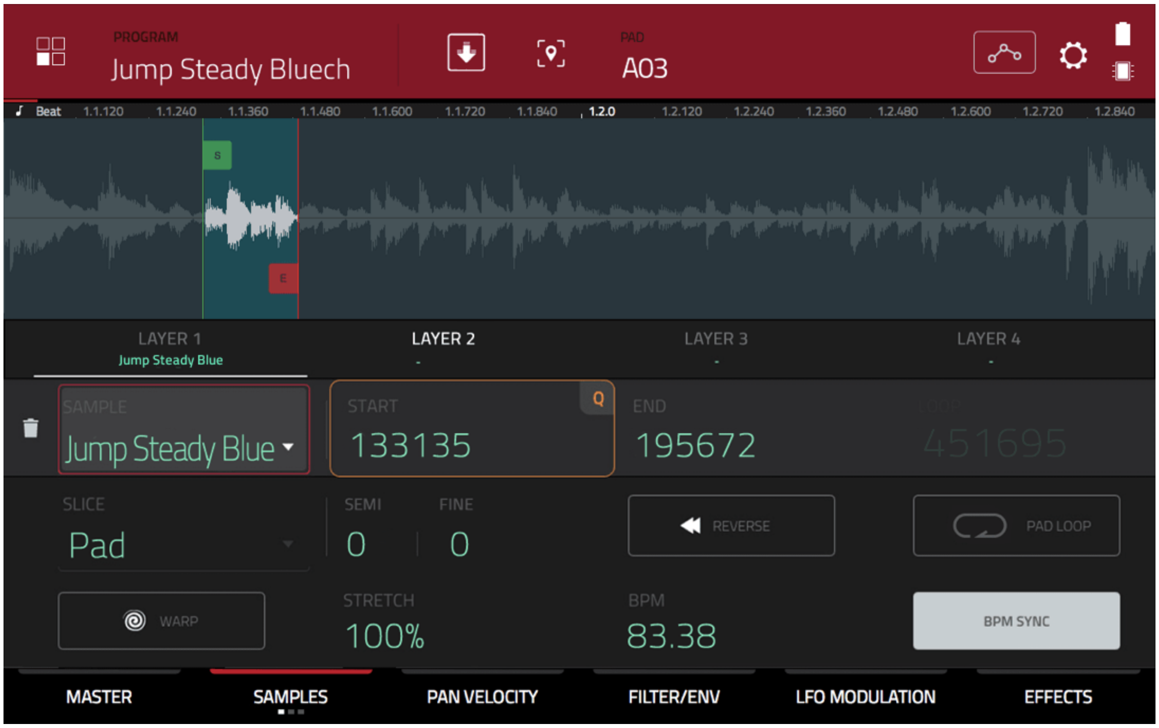
Chopping Using Pad Parameters

Hit **SHIFT** and **CONVERT** and choose '**New Program Using Slices**'. Under '**Slice Type'** select '**Pad Parameters'** and deselect '**With Events'**.



Hit **DO IT** and go to **MAIN** and assign our newly chopped program (‘**Jump Steady Bluech**’) to **track 1** in the **empty sequence 1** so that it becomes the ‘active’ program. Have a play around and you now have all your slices assigned to pads. So what’s different this time?

Go to **PROGRAM EDIT > SAMPLES** and hit pad **A03**:

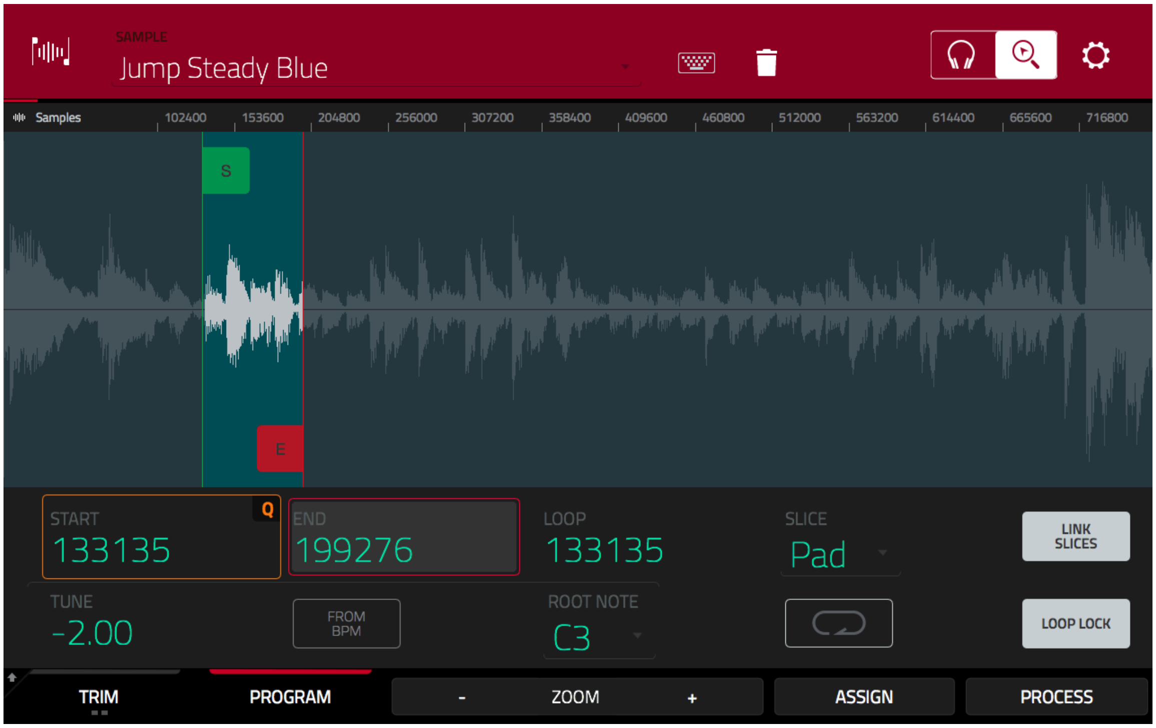


As with standard NDC, each pad is assigned the entire Jump Steady Blue sample, but this time '**SLICE**' is not assigned a specific slice number, it's set to '**Pad**' instead.

How is this different? Well previously our slice regions are set within the original sample itself and remain independent of the program. If we assigned our drum break to another program we would be able to call up those slices within that program as well just by recalling the slice numbers on each layer.

With pad parameters, we've transferred the slices parameters to the program itself. With SLICE set to 'Pad', the START and END points you now see in PROGRAM EDIT > SAMPLES are the 'pad start' and 'pad end' parameters, and are stored within the program itself, not the sample.

Or, as with standard NDC, you can head over to **Sample Edit** and select **PROGRAM Mode:**



As we discovered previously, in Program mode you can directly edit the region assigned to the currently selected pad using any standard editing method.

Using Pad Parameters is still 'non destructive' as you are able to continually make changes to the regions assigned to your pads, and it still only uses a single sample to achieve the end result. However remember that this method does not actually store any of the slices within your original sample – the slice points are entirely stored in your program in the form of those Pad Start and Pad End points.

Experimenting With Your Initial Chops

So at this point you could start moving your chops around and coming up with a mixed up version of your original loop, however I'd prefer to do something far more interesting! The idea here is to really start listening to your current selection of chops and try to pinpoint areas that could form the foundation of a completely new beat – a little hook, a riff, a phrase etc. As we've already chopped this loop into more manageable chunks, the process is a little easier than trying to work on an entire song.

Load up the project file **Blues Chop.xpj** from the chapter **019** folder. This contains the chopped **Jump Steady Bluech** program as made in the previous section, along with the **NDC Slice Kit** I created at the beginning of this chapter.

At this point, take a listen to your existing slices and start to get some ideas about which chops you want to start playing with. The ones that immediately stood out were the chops on pads **4, 7, 8** and **9** so I decided to concentrate on these. As you can hear, they all seem to already 'fit' together nicely, so it's a case of finding a nice order to place them in, and also performing some additional edits of our slices to make them work better in our new beat.

At this point, let's move over to 'something I made earlier'. Go to **sequence 1 (Beat)** and press **PLAY START**. This is a sequenced drum beat I made on **track 2** (**Drums**), with a tempo of **90 BPM** – I'm using the program '**NDC Slice Kit'** for the sounds.

On **track 1** (**Chops**) I've already assigned my own '**Jump Steady Bluech**' program.

What I want to do now is find the 'natural' tempo of our tuned down chops so I can then adjust the tempo of the drum beat accordingly. One way to do this is to first locate a nice 'tight' chop, i.e. one that has a very tight beginning and simply repeatedly trigger this pad, allowing it to finish playing before you re-trigger it.

Pad **A08** is a prime example, and it's also one of the chops I highlighted previously as a good choice for our new beat. So start continually triggering that while simultaneously tapping the **TAP TEMPO** button at the same tempo. You should see the sequence tempo begin to change to match the tempo you are tapping to.

Tap Tempo Accuracy

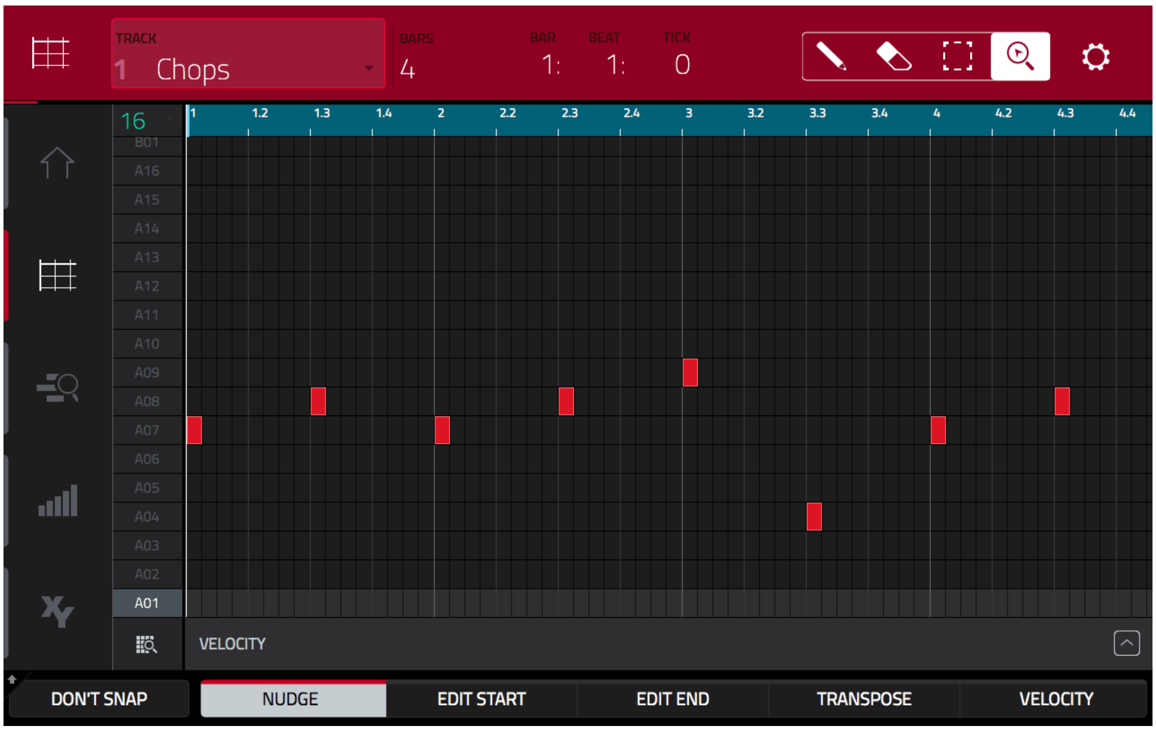
You'll probably find that TAP TEMPO produces quite erratic tempo readings in your sequence, but you can reduce this issue by going to '**Preferences** > **Other'** and change **'Number of Taps'** from '**2 taps**' to '**4 taps'**. This means your tempo is taken from the average of your last four TAP TEMPO hits, instead of the last 2.

The tempo provided by TAP TEMPO obviously does continually vary, so once you've got a decent idea of the tempo you need, stop tapping and enter this tempo into your sequence – I set one of **73.1 BPM**.

If you go to **sequence 2 (Find Tempo)** and press **PLAY START** you can see and hear how just repeating that same region has helped me locate the natural BPM of the beat without too much trouble.

With the tempo set, I now want to get an idea of the 'running order' of the chops, so I spent a little time playing with slices 4, 7, 8 and 9 until I finally settled on this: A7, A8, A7, A8, A9, A4, A7, A8.

That's eight evenly spaced pads over 4 bars, so to lay these down to **track 1**, I programmed these into the **Grid** at intervals running every **2 beats**, using the GRID VIEW techniques covered previously in this book:

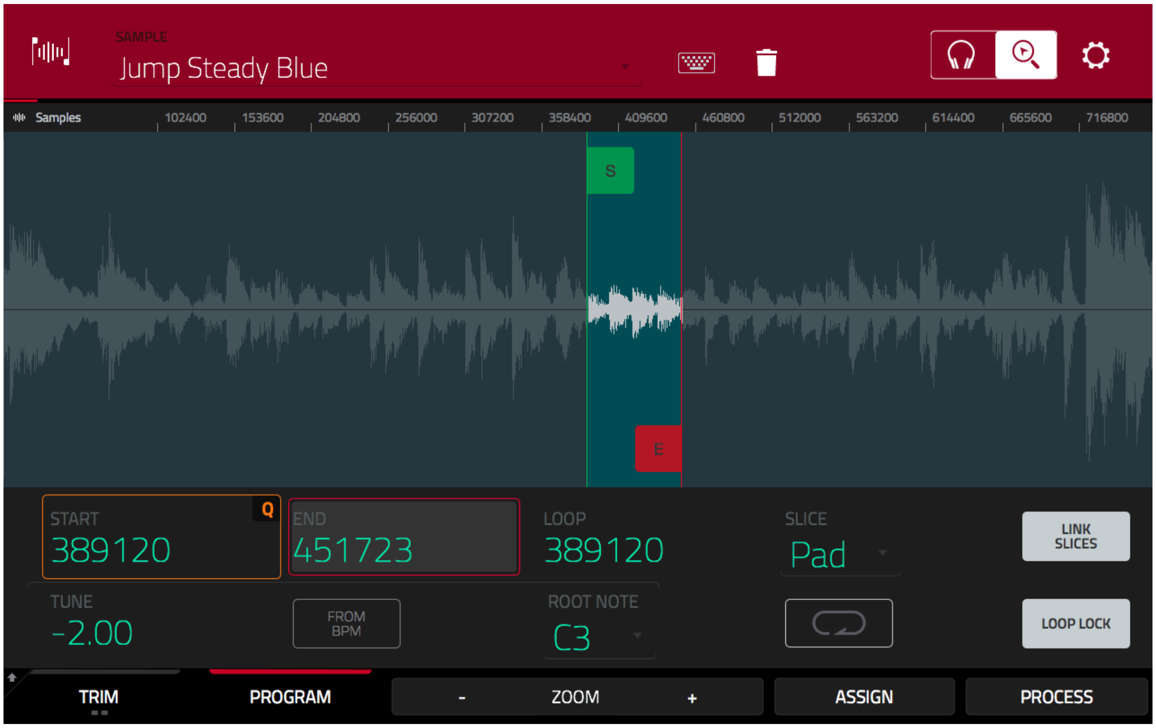


You can hear my version on **sequence 3 (Rough Chop)**. Now press **PLAY START**.

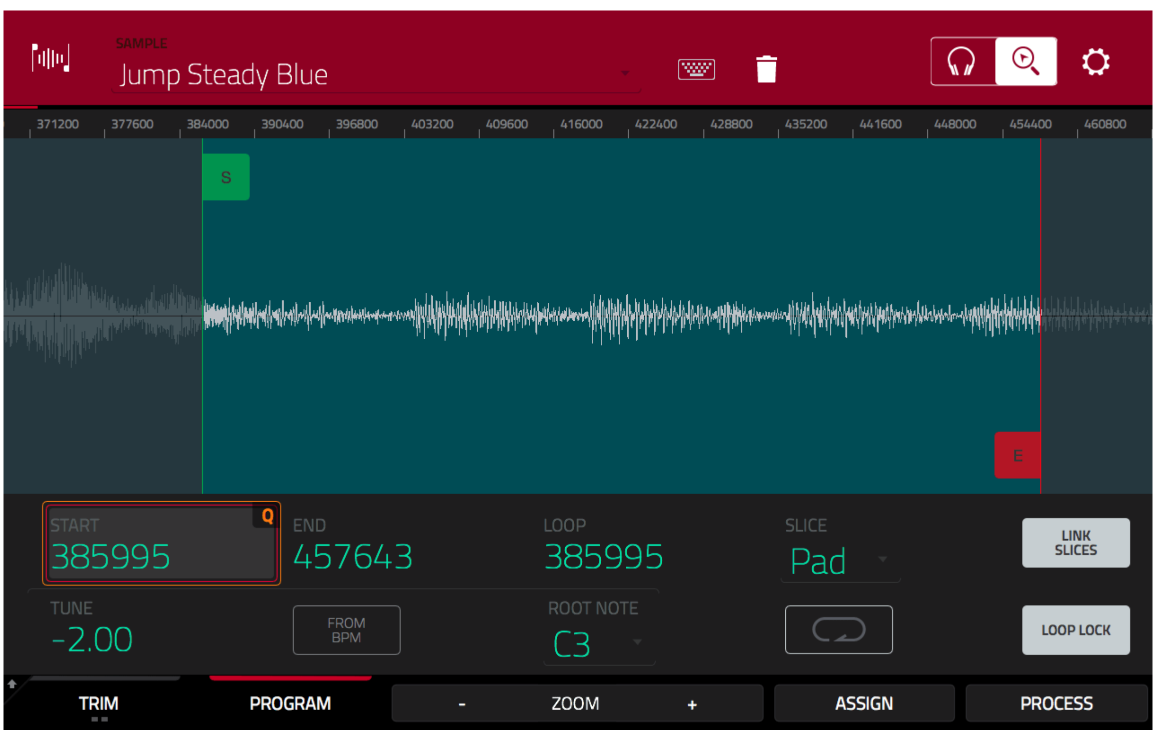
Tightening Your Chops

When we originally chopped this loop into regions, I felt we'd achieved a nice 12 region chop that separated each 1 bar phrase quite nicely. However when you listen to the 'perfectly sequenced' version above, it is clear that we have some tweaking to do. For example, the slice on pad A07 seems a little rushed, while the pad A04 slice is clearly too short and needs extending.

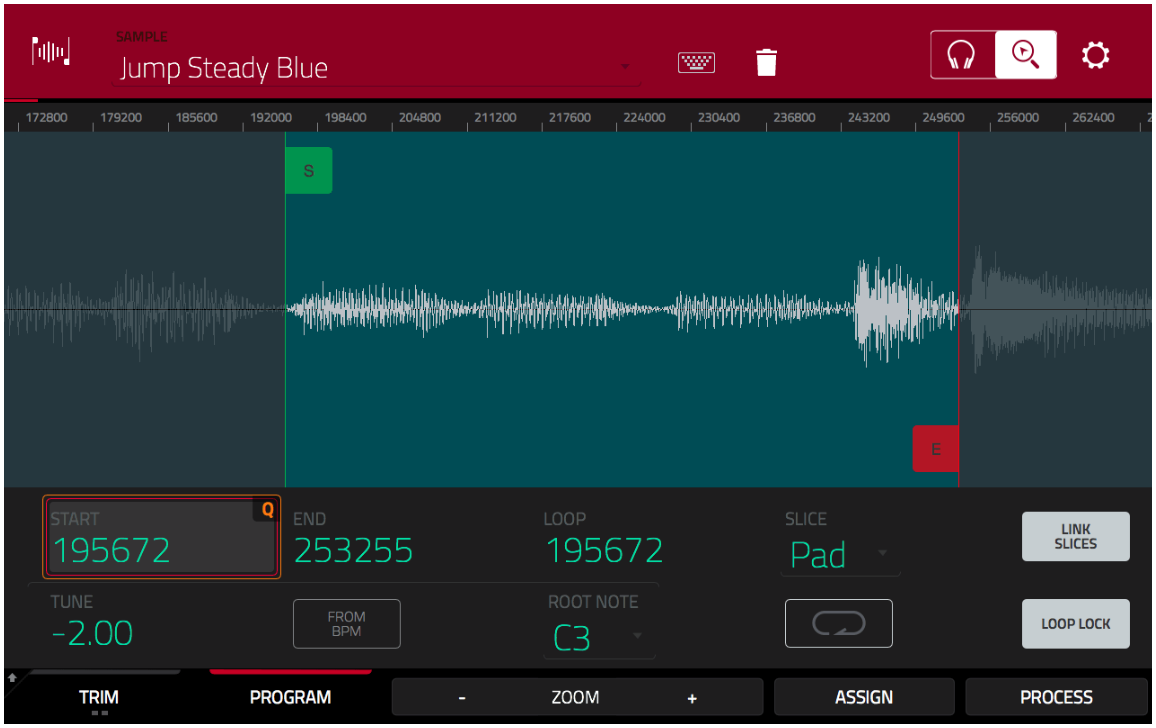
First, make sure you have track 1 selected in MAIN – this ensures that the **Jump Steady Bluech** program is currently selected. Now go to **SAMPLE EDIT** and select the **Jump Steady Blue** sample and press the **PROGRAM** button. Hit pad **A07**.



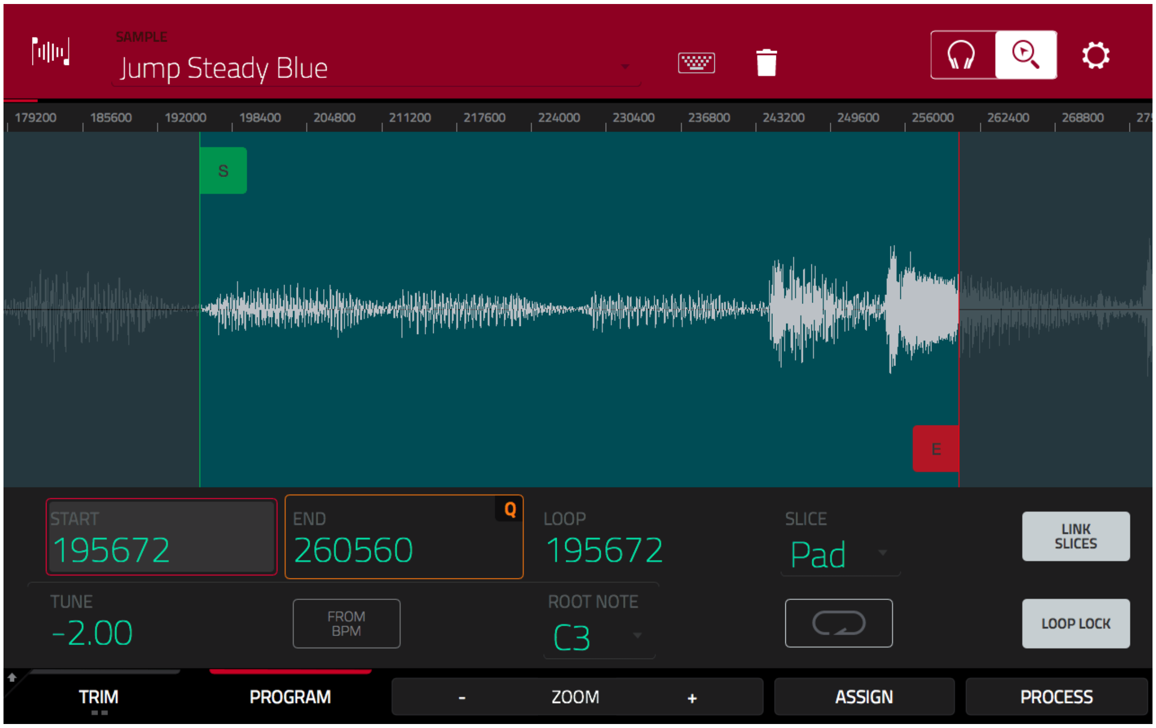
You can hear that the beginning is a little clipped, so let's back off a bit on the slice 7 start point. First, make sure you've set **LINK SLICES** to **OFF**, as we don't want our edits adversely affecting other slices. With **SNAP TO ZERO 'ON'** (**SHIFT > 0 SNAP**), change the start point to **385995**. Preview the edit by hitting pad **A07**.



Now press **PLAY START** and you should hear that all instances of the A07 slice in our sequence fit much better now. Now back in **CHOP > PROGRAM**, select pad **A04**.



This one is too short, it ideally needs to contain that additional note from the start of slice 5, so let's increase the length of **slice 4** accordingly. I set the **end point** to **260560**.



So again, hit **PLAY START** and you'll hear that you've 'filled' the empty space in the sequence.

At this point you can go through all the regions to double check for any oddness while the sequence is playing back - if you wish, rather than making changes in SAMPLE EDIT > PROGRAM mode you can make edits directly in PROGRAM EDIT mode by manually changing the Pad Start and Pad End parameters directly there (remember to press the SAMPLES button three times to show the pad parameters screen).

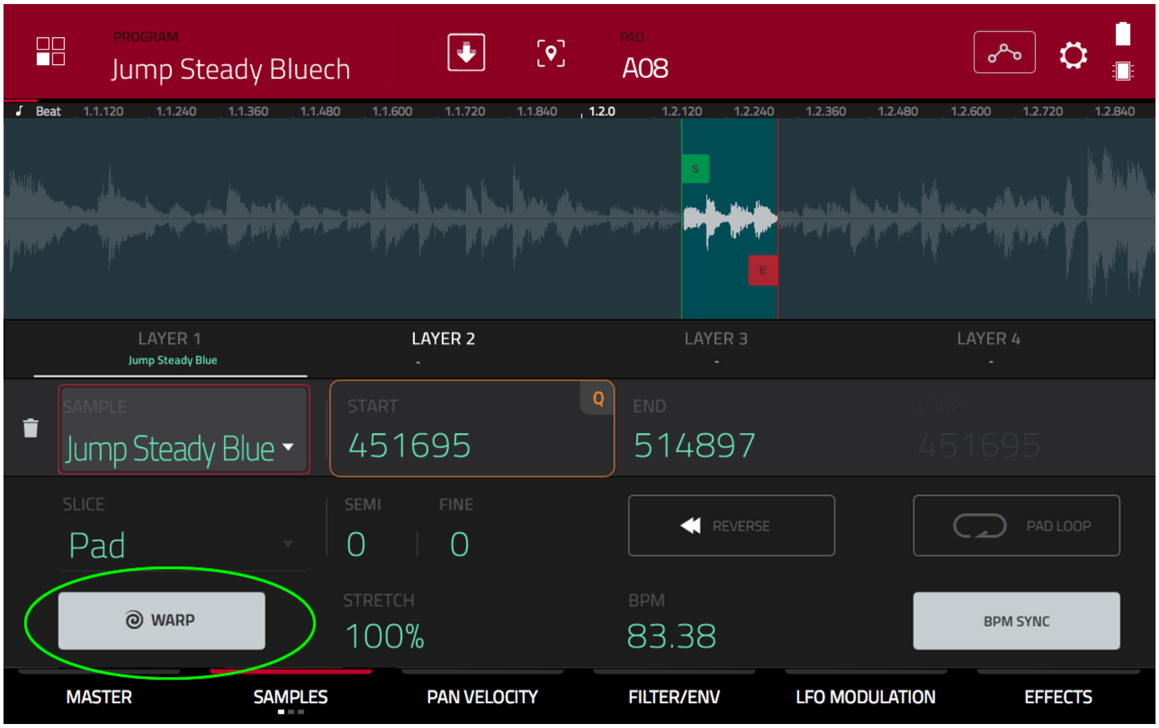
I changed the **start point** of region **8** to **451695** to avoid a small click, and brought the start point of region **9** to **515937** to make it tighter.

Using Warping To Fix 'Choppiness'

While most of the chops are fitting in really well with the beat, I still feel the chop on pad A08 is a little short as there's definitely an audible gap' after it finishes playing which makes the beat sound a bit too 'choppy', especially at the very end of bar 2. It's a similar issue for pad A09 which is used at the start of bar 3.

'Back in the day' we might have extracted this as a standalone sample and time stretched it to make it longer, but we now have a more streamlined solution.

In **PROGRAM EDIT > SAMPLES**, select pad **A08** and engage the **WARP** button:



As you do, you should see the waveform of your slice alter time location a little; this is the 'warp' attempting to stretch our sample to the current sequence tempo. Hit **PLAY START** to allow the sequence to begin looping - while it plays, try turning **WARP** for pad **A08** on and off to hear the difference warping makes to the gap in the audio.

Using WARP 'STRETCH'

Another option here is to adjust the **STRETCH** parameter. This will apply *additional* real time stretch your chop - just tap on the STRETCH parameter for A08 and increase it - the higher the **STRETCH**, the slower the chop becomes. You don't need a lot, in fact I found **101%** gave the chop just the right amount of stretch to make it fit perfectly in the beat.

Repeat this process for pad **A09;** I found a **STRETCH** of **102%** was perfect here.

Finishing Up

Listen to my final version on **sequence 4** (**Jump Steady Final**). Here, I have added a small fill at the end of the sequence, which I created from the very end of the slice on **pad 10**. This is obviously only a very rough song sketch but as you can see, the unique MPC chopping functionality provides you with plenty of different workflow options.

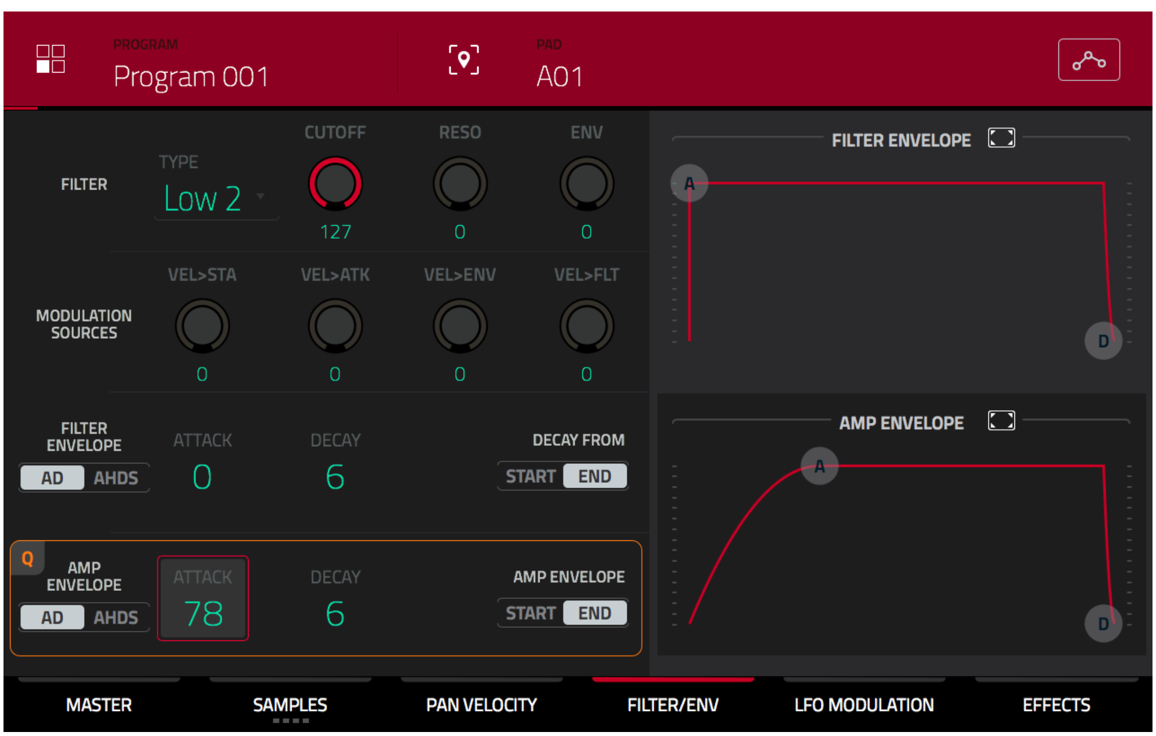
020 Progressive Drum Layering Techniques

In this tutorial we look at how we can overcome 'pad layer' limitations to apply filters and envelopes uniquely to each element of a drum layer – we'll also look at how we can convert our drum layering experiments into standalone samples.

'Sculpting' Together Sample Elements

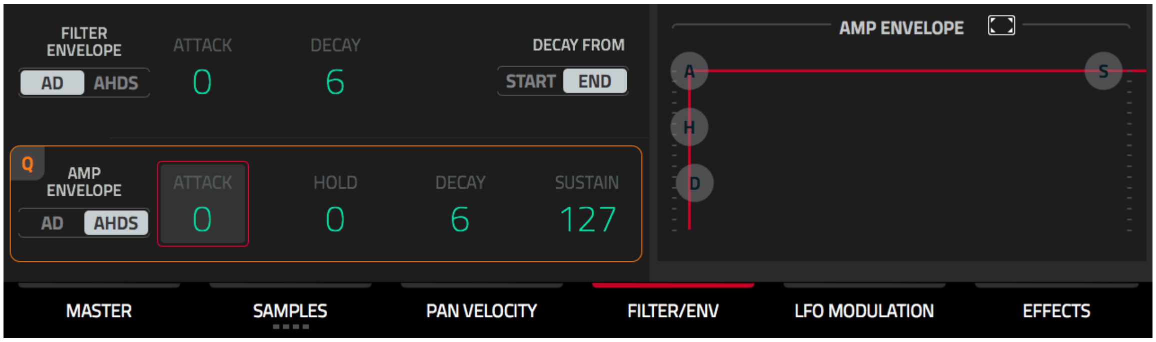
Start with a new project and a blank DRUM program. From the **chapter 020** folder load and assign the **clap_decay** sample to pad **A01** and the **wood-snare** sample to pad **A02**. Preview both sounds – in this tutorial we're going to layer these sounds together, however we're only going to combine specific portions of each sample.

Select the clap on pad **A01** and go to **PROGRAM EDIT > FILTER/ENV**. With this clap I want to isolate only the decay portion of the sound. Tap on the **AMP ENVELOPE ATTACK** parameter and while continually previewing pad **A01**, turn your data wheel anti clockwise until the sharp attack of the clap has completely disappeared– a setting of **78** should do the trick.



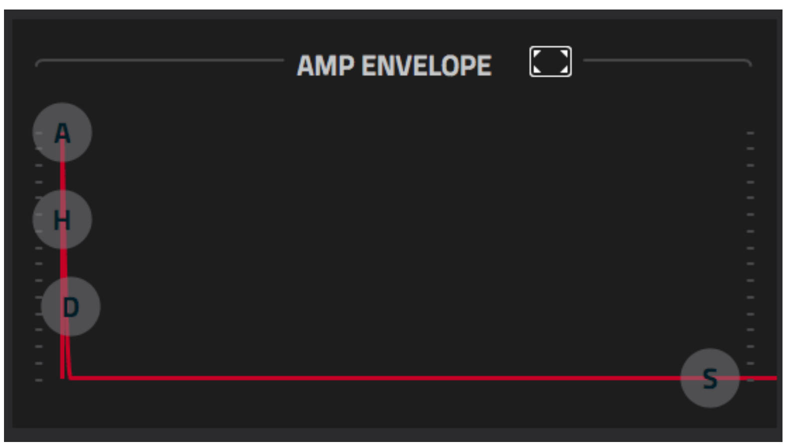
So here on pad **A01**, we have created the 'decay' part of our layered sound.

Now over to the snare on pad **A02**, where I want to keep that lovely attack and lose the decay (so I can replace it with the decay from the clap). This time, let's use the other Amp Envelope available to us – the **AHDS** envelope. To access this, we need to change the Amp Envelope '**type**'. Under **AMP ENVELOPE**, select the **AHDS** envelope



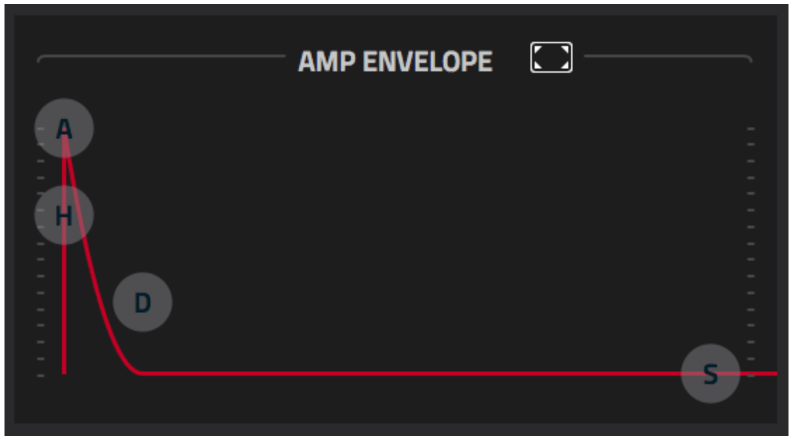
Take a look at the waveform. We now have four parameters to adjust. **ATTACK** is the same as we've seen in the AD envelope. We then have **HOLD** (**H**), **DECAY** (**D**) and **SUSTAIN** (**S**). Let's see how these all work to allow us to 'shape' our snare.

Firstly, turn the **SUSTAIN** value down to zero:



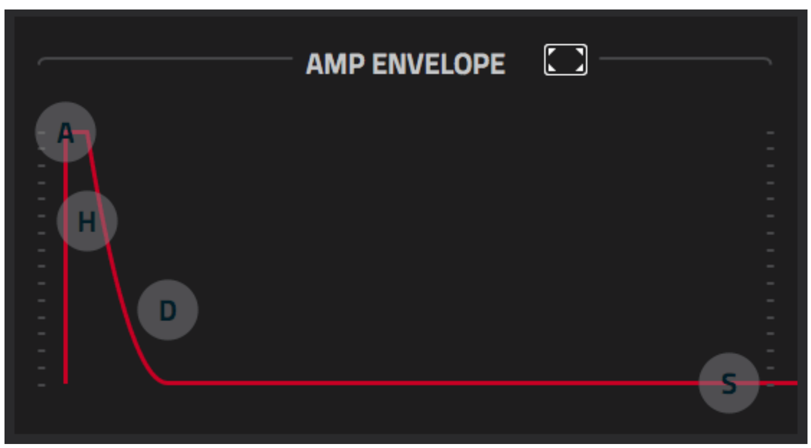
If you listen to pad **A02** it's now nothing but a short click. By giving the sample no sustain whatsoever we've removed pretty much all the audio from our sample apart from a very brief initial attack.

Now turn the **DECAY** to a value of **59**:



If you preview **A02** now, you can hear a bit more of the attack. As you can see from the envelope graph, the decay is controlling the way the sound makes its way to the sustain portion of the envelope. The higher the decay value, the longer it takes for the sound to reach the sustain you've set. With a sustain of zero, the decay portion is basically controlling a standard fade out, similar to the decay of the AD envelope.

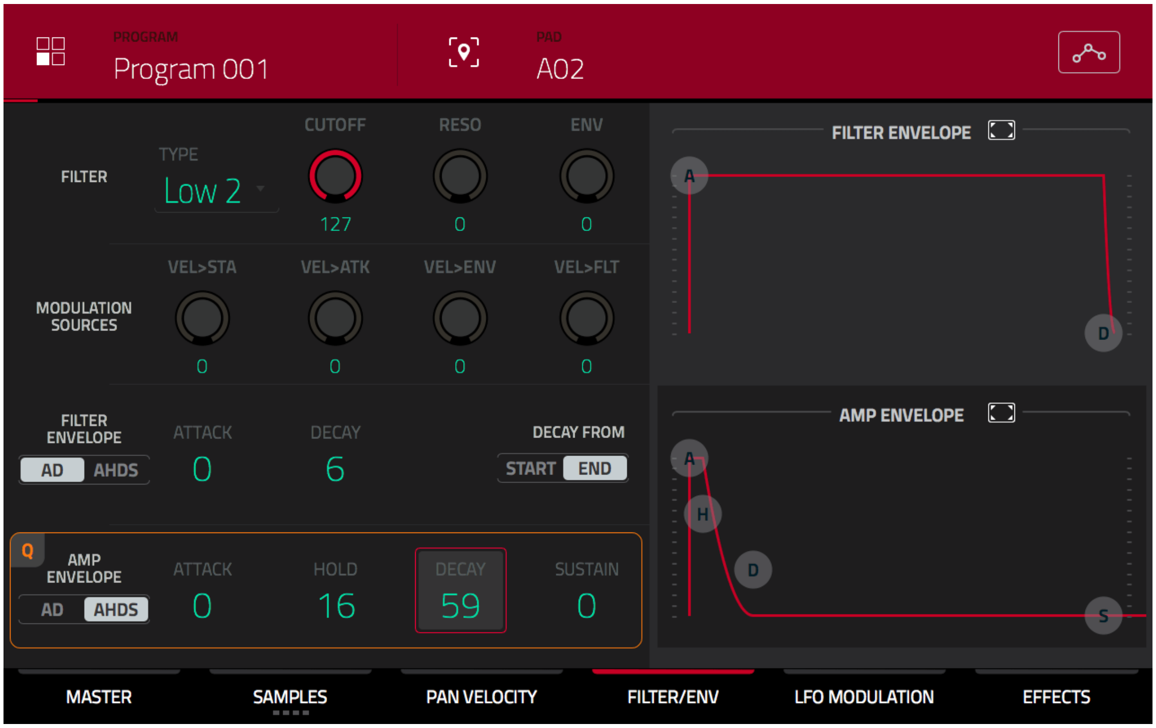
Now I want to hear a bit more body from our snare, so gradually increase the '**HOLD**' value to **16**.



As you can see from the graph, the hold value is literally 'holding' the initial volume before letting the decay portion take over. The higher the hold value, the more of the sample will play before it enters the decay portion of the envelope. Preview **A02** and you should hear that by 'holding' the sample for a while, it's allowing us to control the amount of snare 'body' that will be heard before the sound fades out in the '**D-to-S**' portion of our **AHDS** envelope.

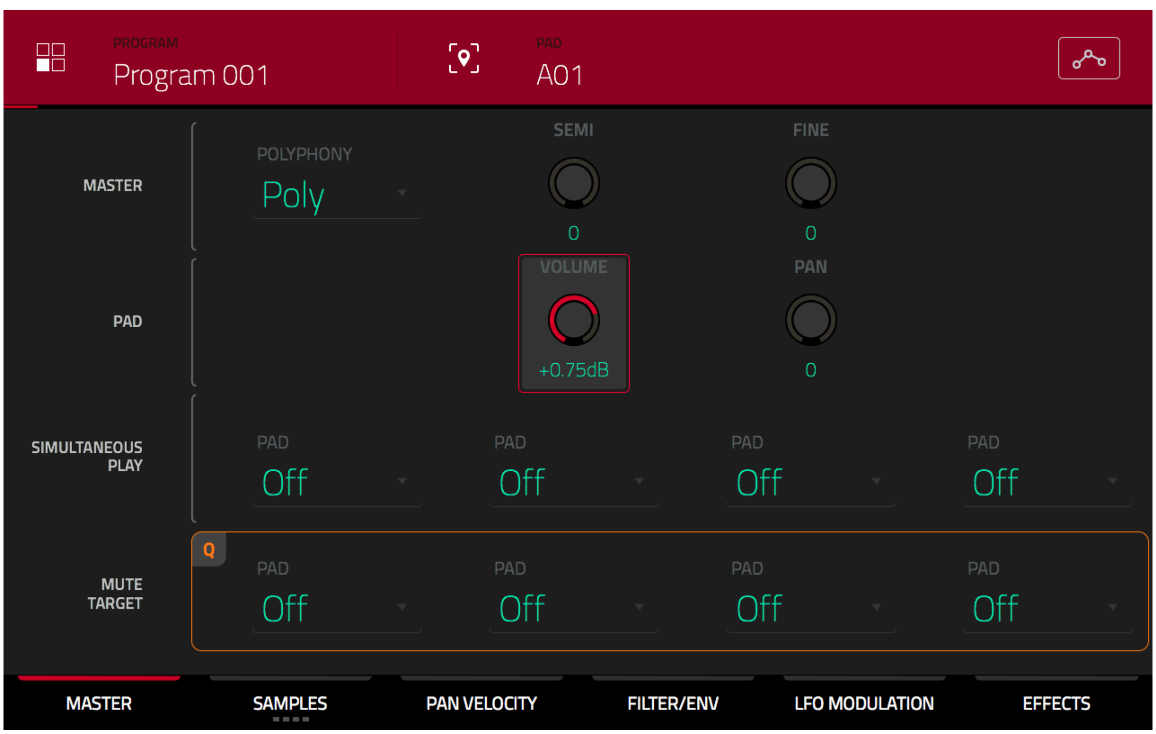
This is pretty sweet as it gives you a very nice way to precisely 'mould' your sounds. Our new snare is now tighter sounding and has minimal decay – it's mostly attack and body.

If you wanted more body, simply increase the **HOLD** value a bit more. If you want a longer decay, increase the **DECAY** value.



Now head over to **MASTER** and select pad **A03**. Under **SIMULTANEOUS PLAY**, choose **A01** and **A02**:

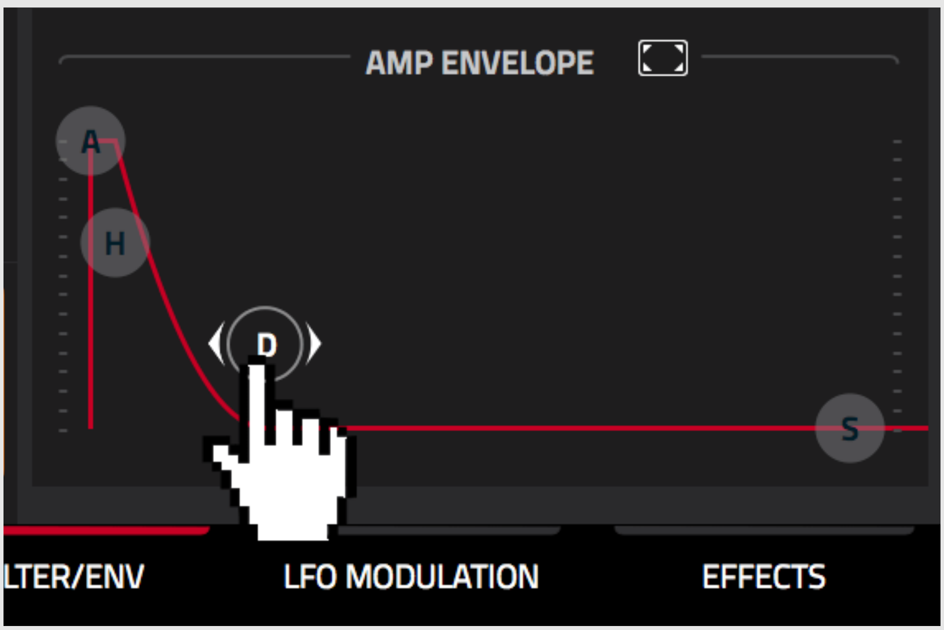
Now preview pad A03 to hear our newly 'painted' snare. If you wish you can further tweak each 'layer' to taste – for example if you want the clap decay element to be more prominent, select **A01** and in **MASTER** increase the **PAD VOLUME** setting.



This method is a more refined method of layering compared to just slapping 2 or 3 sounds on top of each other. And there's no need to stop at snares and claps - try the same technique with kicks. Take a booming bass drum and layer it with a thin old vinyl kick - try shaving off the attack on the booming kick and maybe take a little decay off the vinyl kick and you should get a nice crunchy kick drum with plenty of low end.

Directly manipulating the envelope waveform

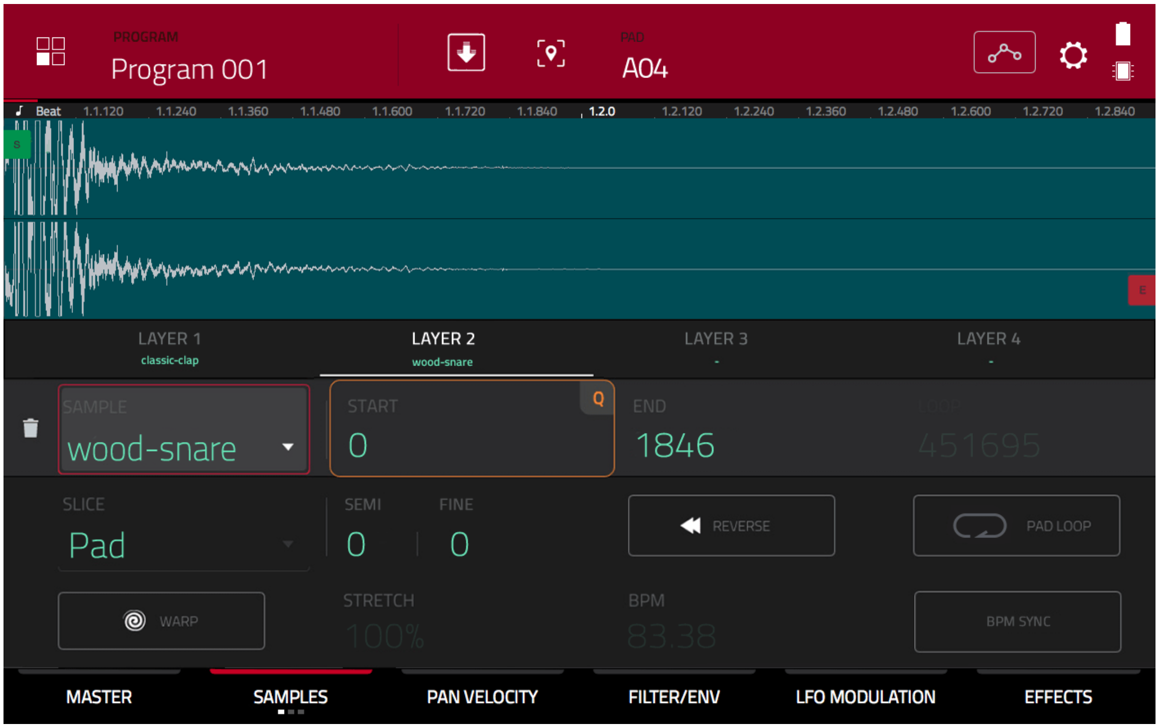
An alternative method of setting the envelope parameters is to use your finger to simply drag the envelope points:



Layering With Pad Parameters

We also have another way to 'paint' with sample layers, this time using the layer-specific '**Start**' and '**End**' program parameters.

Load the **classic-clap** sample into your project and assign it to **layer 1** on pad **A04**. Now assign the **wood-snare** sample to **layer 2** of pad **A04**. Set the '**SLICE**' parameter for both layers from **All** to **Pad**.



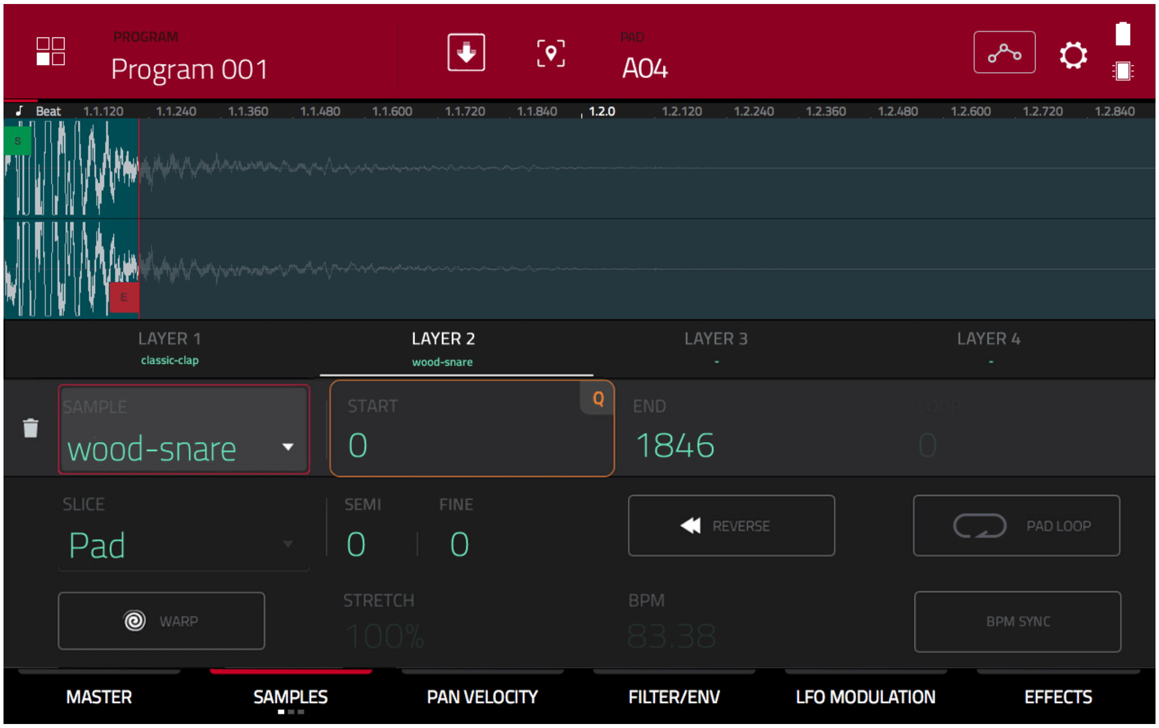
As we've seen previously, when we set the SLICE parameter to 'Pad', the 'START' and 'END' points assigned to each layer on a pad are independent of the 'master' start and end point of the sample itself; this is, they are unique to this particular *instance* of that sample.

Here I am going to perform a similar task to the envelope 'sculpting' we did earlier, but this time I'm going to isolate the desired portion of each sample using the pad start and pad end points on each layer, thus negating the need for using simultaneous play settings over multiple pads.

First let's set up the wood-snare sample. As we don't have a 'layer solo' option, briefly head over to Go to **SAMPLES tab 2** and set the **LEVEL** for '**layer 1**' to **0** – this way when we preview pad A04, we are only hearing layer 2, the **wood-snare.**

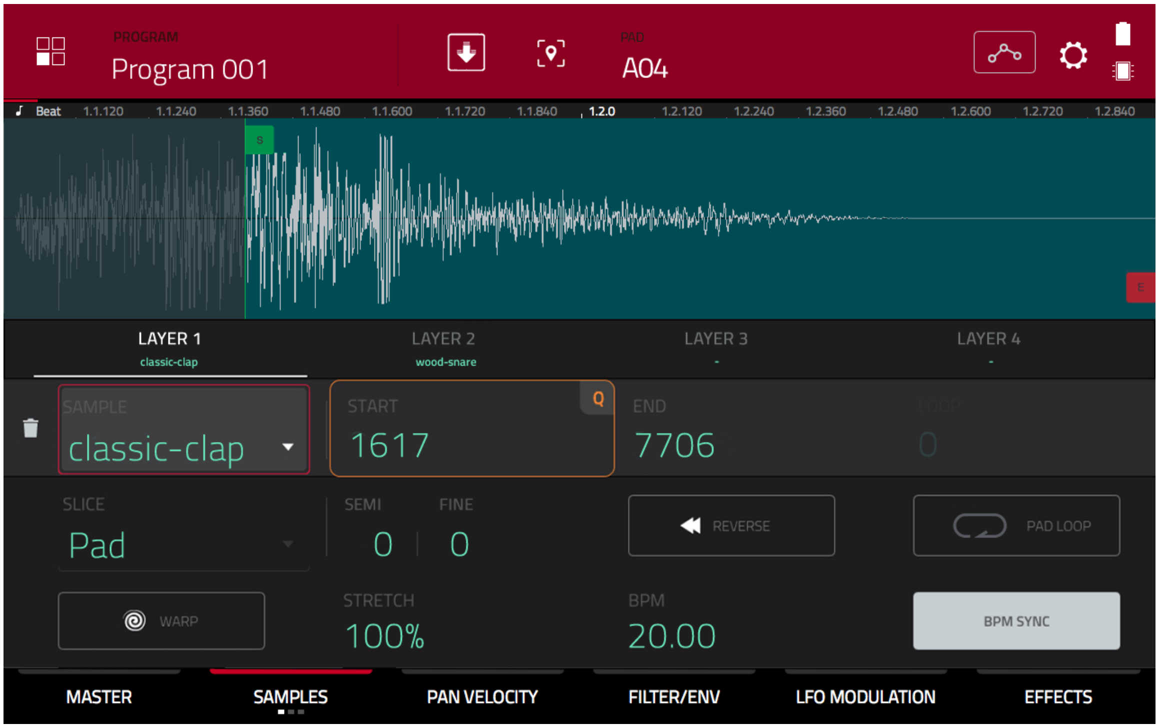
Now return to **SAMPLES tab** 1, **select LAYER 2** and start reducing the **END** value down from **20571** – use either the tap and data wheel method, the **Q-Links in column 2**, or simply drag your finger on the red END point flag to the left. Remember to hold down **SHIFT** for more accurate single digit changes.

As you reduce the pad end value, keep on hitting pad **A04** to hear the effect this is having. We just want to isolate only the initial attack portion of this sample - I found a value around **1846** worked well.



Now briefly return to **SAMPLES Tab 2**, set the **layer 2 LEVEL** to **0** and then return the **LEVEL** for **layer 1** to **127**. Now when you preview pad **A04** you can only hear layer 1, the clap.

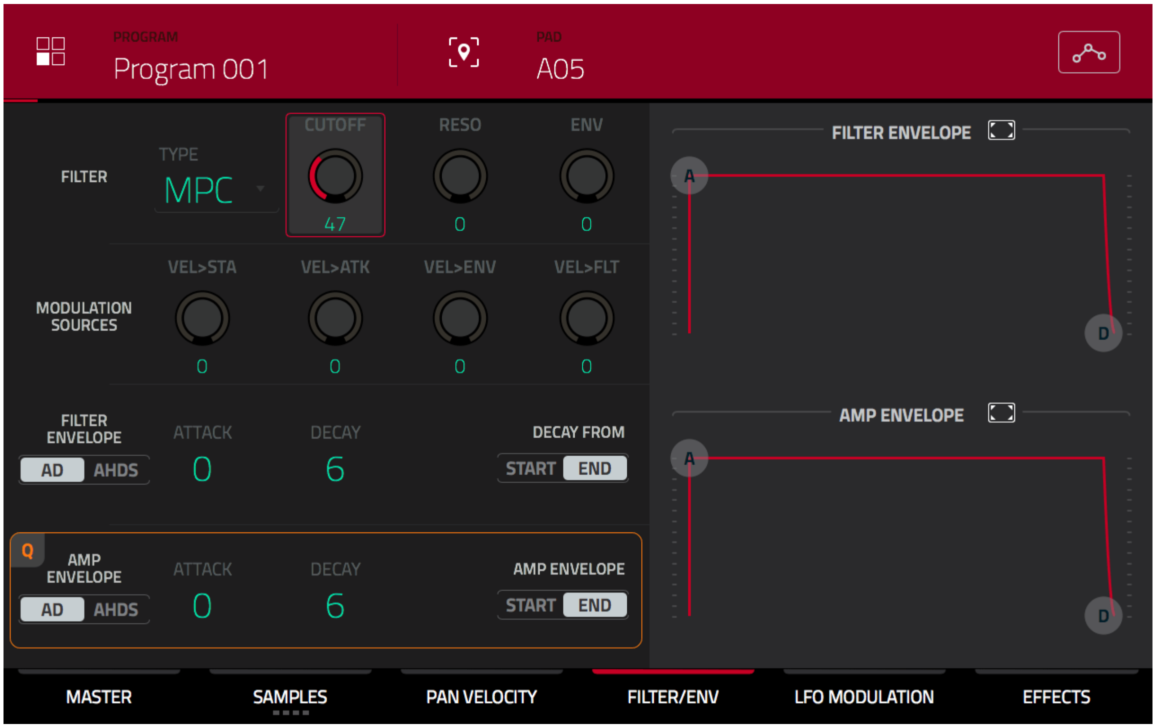
Let's repeat the procedure for the **classic-clap** on **layer 1** to isolate the 'decay' portion of this sample. To do this, start increasing the **START** value. Once again, as you increase the value, keep previewing by hitting pad **A04** until everything sounds good. I used a value of **1617**.



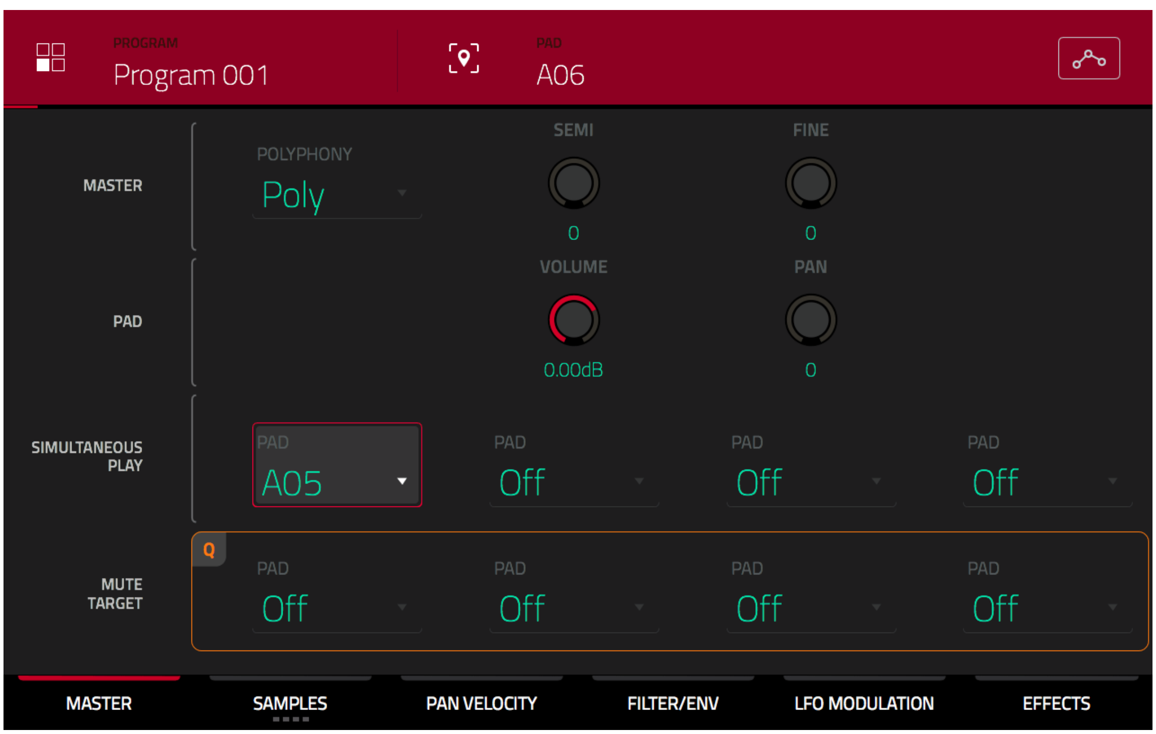
All that's left now is to 'mix' the two layers together, which can be done in the **SAMPLES tab 2** screen using the **LEVEL** settings for each layer. An interesting way to do this is to set the **layer 2 (snare) LEVEL** to **127** and **layer 1 (clap)** to **0**, then while previewing continually on pad **A04**, start increasing the **LEVEL** of **layer 1** gradually bringing up the level of the clap decay until it mixes perfectly with the snare attack. I set the clap layer level to **110**.

Applying Filters to Individual Layers

Here's a great way to fatten up a single kick sample. On pad **A05** load and assign the **deep-kick** sample to **layer 1**. Now go to **PROGRAM EDIT > FILTER/ENV** and set a **FILTER TYPE** of **MPC** – this is an emulation of the MPC3000 low pass filter. Now reduce the **CUTOFF** value to **47** to give us a very low and booming kick sound.



Now assign that same kick to pad **A06**. Go to **MASTER** for **A06** and set it to **SIMULTANEOUS PLAY** with our filtered version on **A05.**



You've now got a significantly more powerful kick with huge amounts of bottom end, but still retaining the higher frequency elements found in the original kick.

And because you are filtering one of the layers, you avoid any possible phasing problems that can sometimes occur when layering identical samples on top of each other.

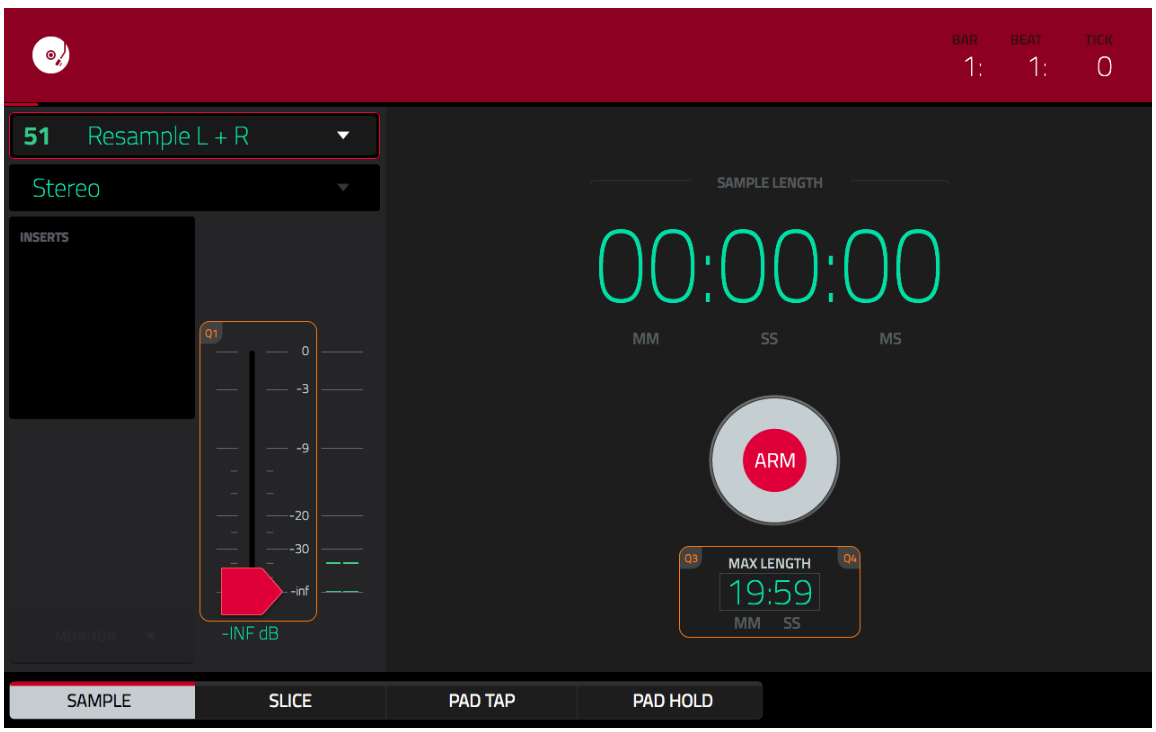
Converting Your Layers Into Standalone/Portable Samples

Once you've created your newly designed sounds, you can of course leave them set up in your program and use it 'as is'. However, often you'll want your creation to be a lot more portable so it's commonplace to convert your new creations into 'standalone' samples.

There are currently two ways we can convert a pad into a standalone sample; we can '**flatten**' the pad or we can **resample** it. Let's look at resampling first.

Resampling Your Pad

Go to **SAMPLER** mode from the main **MENU** page:



This time you can see that I have set the 'input' to '**RESAMPLE L+R**'. This simply records the 'internal' audio being created by your MPC before it is sent out to the audio interface. Try it out – press pad **A03** to hear the 'sculpted' snare we made previously. You can see the recording meter rise as the sound plays – changing the REC GAIN dial will do nothing as we are not using the actual recording inputs of our interface.

You'll probably find that the sample is clipping (the level meter is heavily in the red). Sometimes this clipping can work out well, but often it's better to record the sample with some 'breathing space' as we can always make the sample hotter during post production.

Unfortunately there's no way to change the input gain when resampling, so my usual workaround is to head over to **CHANNEL MIXER > Programs** and temporarily reduce the **LEVEL** for your program - remember to use the level meter on this screen; keep hitting pad **A03** while you reduce the **LEVEL** value and stop once you see the LEVEL stop clipping in the red. A setting of **-7.00dB** works well.

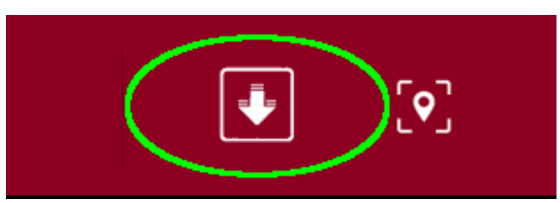
Now head back to the SAMPLER page, press '**ARM**' and then press pad **A03**. This should trigger the recording process. Press the stop button to stop recording – preview the sound with '**PLAY**', name it '**Sculpted Snare**', set **ASSIGN TO PAD** as **A13** and press **KEEP**. If you did reduce the program level, remember to change it back.

Now, this newly created sample will be just like any sound we record through the SAMPLER page, hence it will at the very least need 'topping and tailing' in SAMPLE EDIT mode using the techniques we've discussed previously in the book.

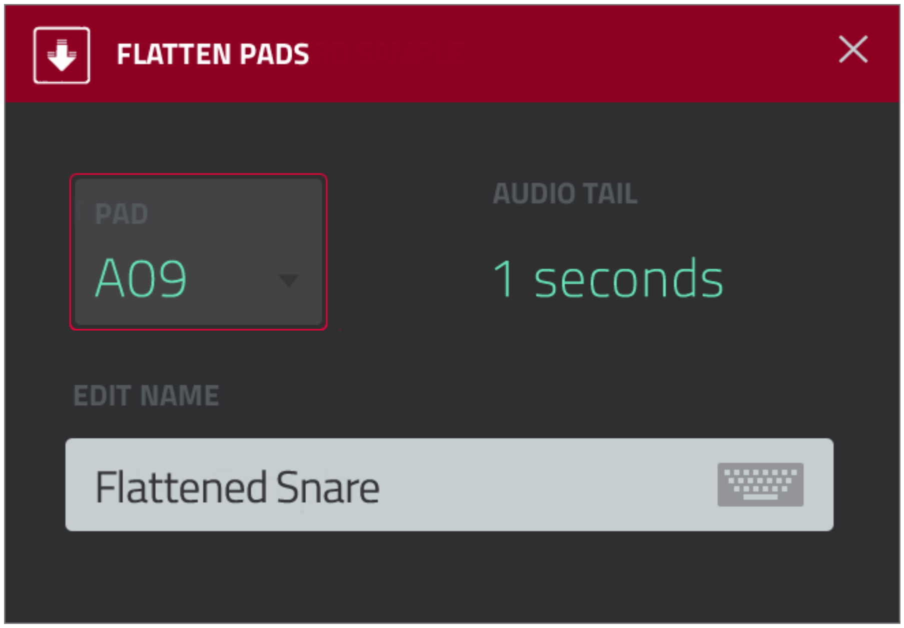
Flatten Pad

The other way to convert a pad into a standalone sample is to 'flatten' the pad. Similarly to resampling, the flattening process creates an 'audio' snapshot of the current state of your pad, however the difference is that flattening is a *destructive* process.

Let's flatten the layered snare we created using pad parameters on pad A04. I'm going to copy pad A04 to pad A09, but that's entirely optional.To flatten your pad, first hit the '**Flatten Pad**' icon at the top of the screen:

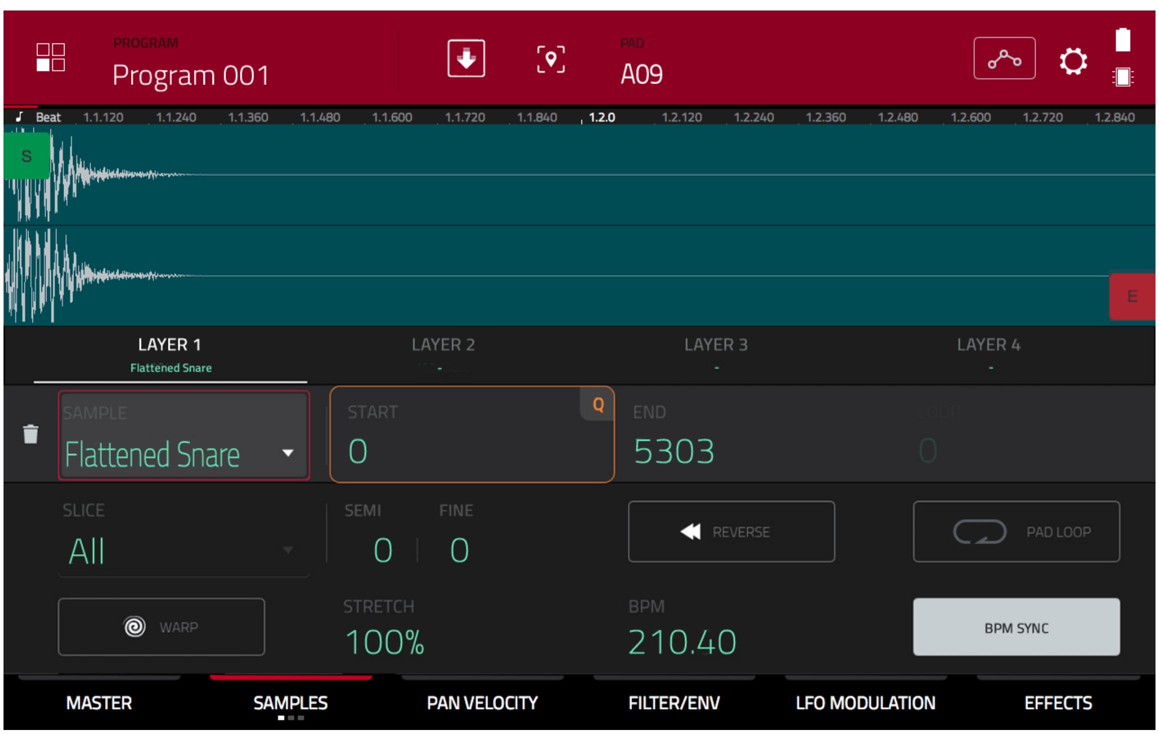


This brings up the **FLATTEN PADS** screen:



This confirms that you are flattening pad **A09**. The **AUDIO TAIL** setting is useful for when you have a delay type effect on your pad as this will extend the time the MPC records the pad. We can leave this at '**1 seconds**' for our current pad.

Finally you can set a custom name for the resulting sample such as '**Flattened Snare**'. Hit **DO IT**.



As you can see, there is now a sample called '**Flattened Snare**' assigned to **LAYER 1** of pad **A09**. All other pad parameters have been reset; there's no sample on the second layer, no pad parameters, and so on. This is why I refer to 'Flatten Pad' as a 'destructive' process.

Of course initially copying the pad is not required, in fact in some cases the reason for flattening a pad is to free up program resources so you'll not want to have the original pad still in the program. Just remember that all those pad parameter settings are lost after flattening - however, you can use UNDO if you change your mind.

The main advantage of flatten pad is it's *really* quick - two taps and you have made a standalone sample from your pad.

Remember after creating your new 'standalone' sample (using whichever method you choose), you are the free to continue applying more processes to that sound. For example after applying the maximum 4 insert effects to a pad you could flatten it and then apply even more effects to the new 'flattened' sample. Using flattened samples is also a great way of applying multiple filters to a sound.

Load up the program **Advanced Layering.xpm** to see my version of the techniques used in this tutorial. This program also contains re-sampled and flattened snare sounds on pads A09 and A13.

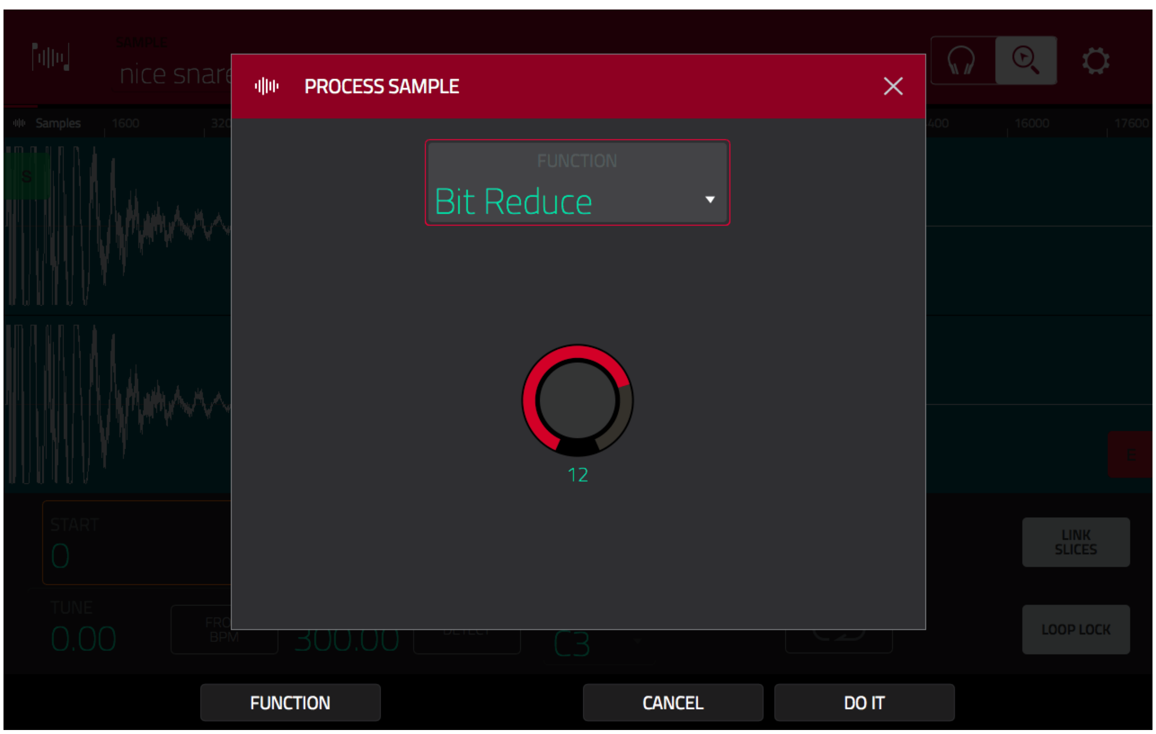
021 Sound Design Tricks

The MPC may not be the world's most advanced sampler, but it still provides us with plenty of sound design tools. In this chapter we'll take a look at some classic MPC sound design techniques

Emulating Vintage Sampler 'Crunch'

One of the most desirable drum sounds for a beat maker is the classic 12 bit sound of samplers such as the Akai S900/950 and the Emu SP12. While you are never going to emulate this sound *exactly* (as much of it is down to hardware components in these old samplers), you can certainly try reducing the number of 'bits' in your sample, as this can give you all sorts of interesting variants on your original 'clean' sound.

From the **chapter 021** folder, load the sample '**nice snare'** and open in in **SAMPLE EDIT**. Preview with **pad 10**. Hit the **PROCESS** button at the bottom right of the MPC screen and select '**FUNCTION**: **Bit Reduce**':



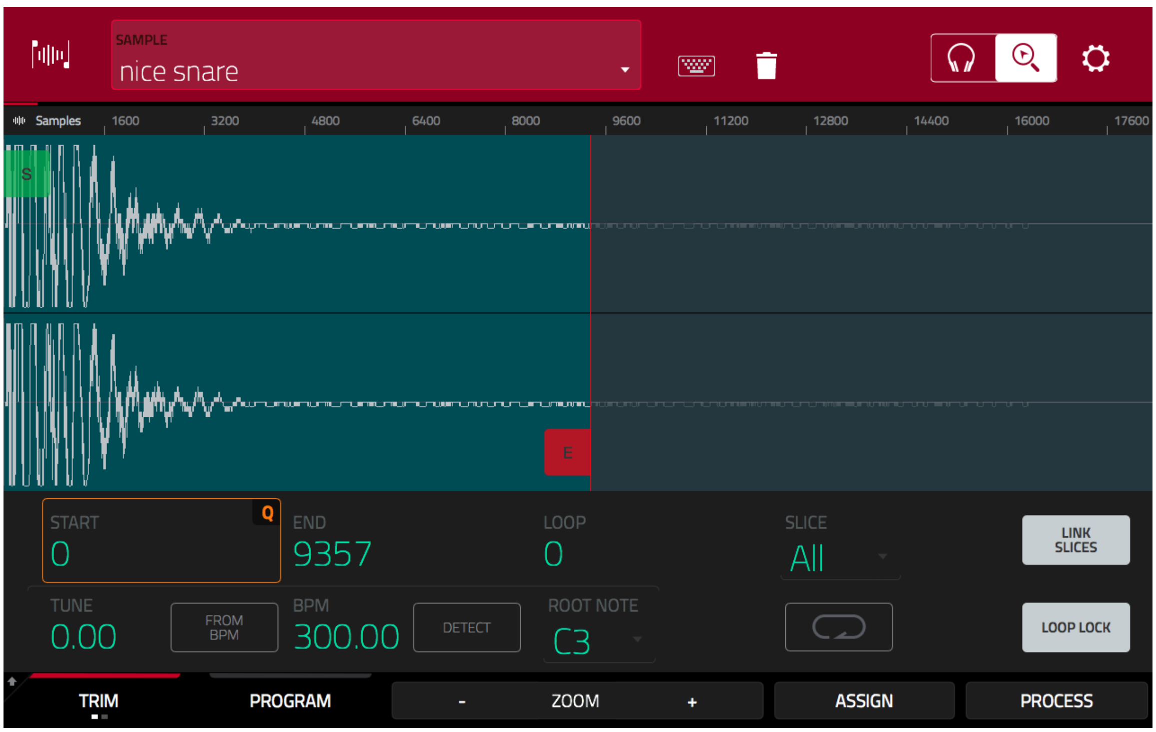
The **'Bit depth'** setting here refers to the bit rate of the sample - we'll leave this set at '**12'** for the moment. Remember that it is usual to work with 16 or 24 bit samples in an MPC, so anything with a lower bit rate will have reduced quality.

To perform the bit reduction, press **DO IT**. By reducing to 12 bit, you'll not initially hear an *obvious* change in sound, as the reduction to 12 bit is quite subtle. It does degrade your sound by a small amount, and will tend to give your drums a very slight bit of extra 'grit' without making obvious changes to the sound.

However to take matters further, let's try **4 bit** reduction. First let's return our sample to its original **16 bit** state by pressing the **UNDO** button. Now select **Process**, **BIT REDUCE**, and set the '**Bit Depth'** dial to **4**, and press **DO IT**.

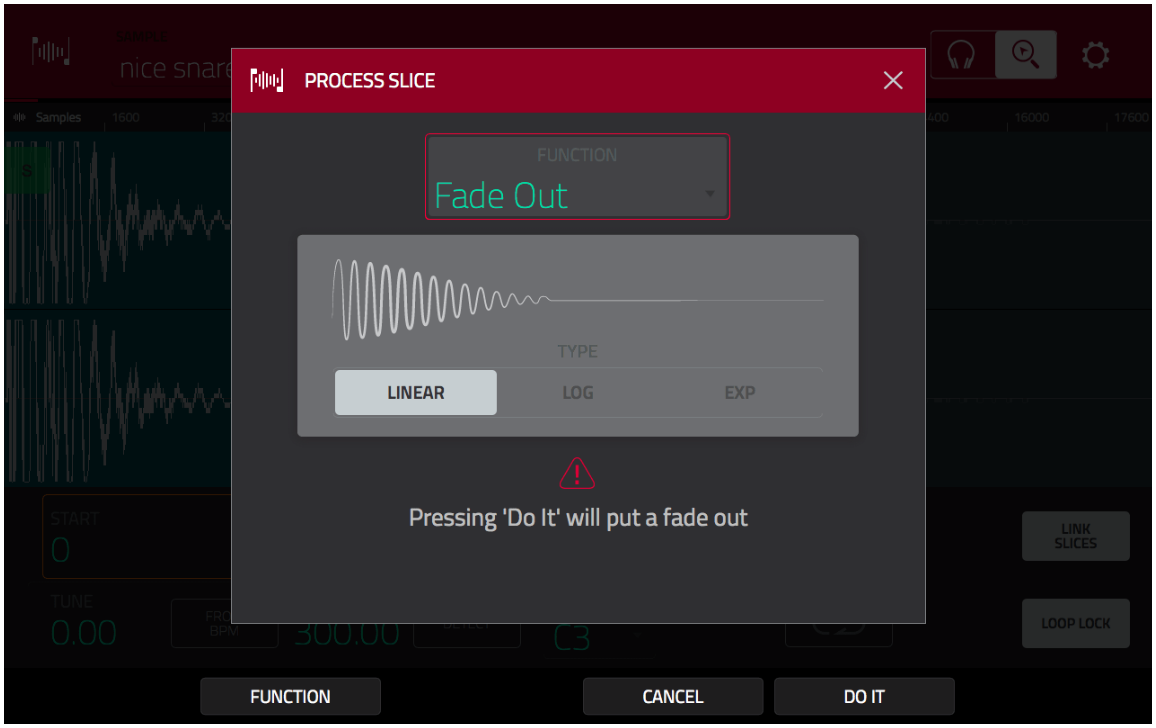
At 4 bit, the bit rate change is more obvious, you'll hear it affect the reverb tail of your sample which will become crackly and very 'bitty'. This can be heard much more clearly if you listen with your headphones. To counter this problem, let's utilize some more sample processing options.

In the main **SAMPLE EDIT** screen adjust the **END** point to approximately **9400** (it doesn't need to be completely accurate). Preview with **pad 10**.

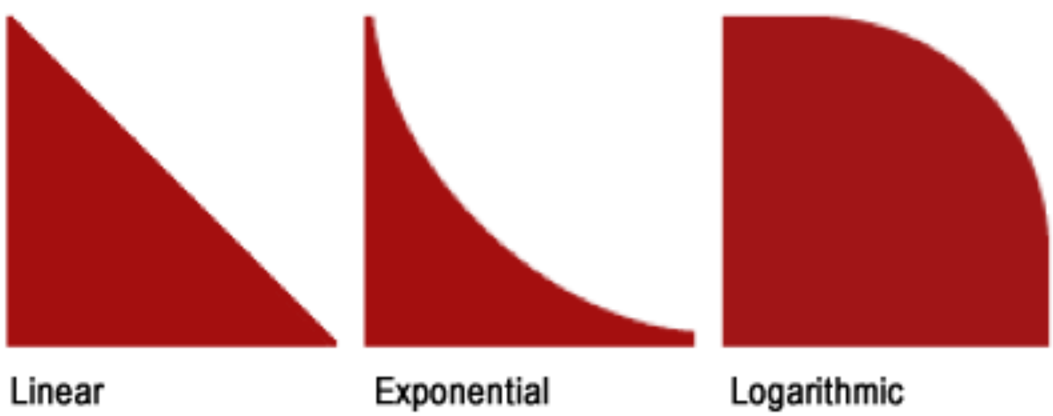


This has removed a lot of the distorted snare tail, - however this abrupt end to the snare isn't great, so we need to create some kind of decay. To tackle this issue, we could simply adjust the Amp Envelope when assigning this sound to a pad in a program, i.e. use the 'Decay' value to apply a fade out to the sample.

However in SAMPLE EDIT we can also apply a *permanent* fade out. Hit **Process** and select **FUNCTION: Fade Out**':



As the name suggests, this editing function will fade our sample out by gradually decreasing the volume to zero by the time it reaches the end point (there's an opposite acting function '**Fade In'** which acts similarly, but fades the sample in from the start point). There are three **types** of fade out possible; **Linear**, **Log** (logarithmic) and **Exp** (exponential) and each one reduces the volume to zero using a slightly different 'shape':



A **linear** type fades out the sample very gradually, and **exp** fade starts gradually, speeds up in the middle and ends more gradually, and **log** starts fading quickly, slows in the middle, then finishes quickly. As we have the **UNDO** feature, we can try out all three FADE types to see which one gives the best results. So, try each one, pressing UNDO after previewing each one.

After trying out all three fades, I felt that **Linear** gives the best results - certainly in terms of providing a snare sample with no distorted or 'hissy' tail. So the end result is a nice 'gritty snare sound. Hit **PROCESS > DISCARD** to remove the unwanted sample data at the end of the snare.

Saving Individual Samples

There's currently no way to save individual samples within SAMPLE EDIT, but if you head over to **MAIN** and hit the **pencil icon on the PROGRAM row**, there's an option to **SAVE CURRENT SAMPLE**. This should save the last sample 'accessed' by your MPC, so as long as you go directly from SAMPLE EDIT to SAVE CURRENT SAMPLE without hitting any pads along the way, the MPC should offer to save the sample you were last working on in SAMPLE EDIT.

Otherwise, this option will save the sample assigned to the most recent pad hit. I do however find this feature doesn't always work as expected!

Load up my version, '**nasty snare'** from the **chapter 021 folder** (or just preview it in the BROWSER).

How low can you go?

You can take bit reduction down as low as 1 bit – at this point you can barely make out it's a snare! You can try this and just UNDO if you don't like it – while it may not sound like a snare anymore, you might still find this usable for one of your projects, perhaps as a layering element.

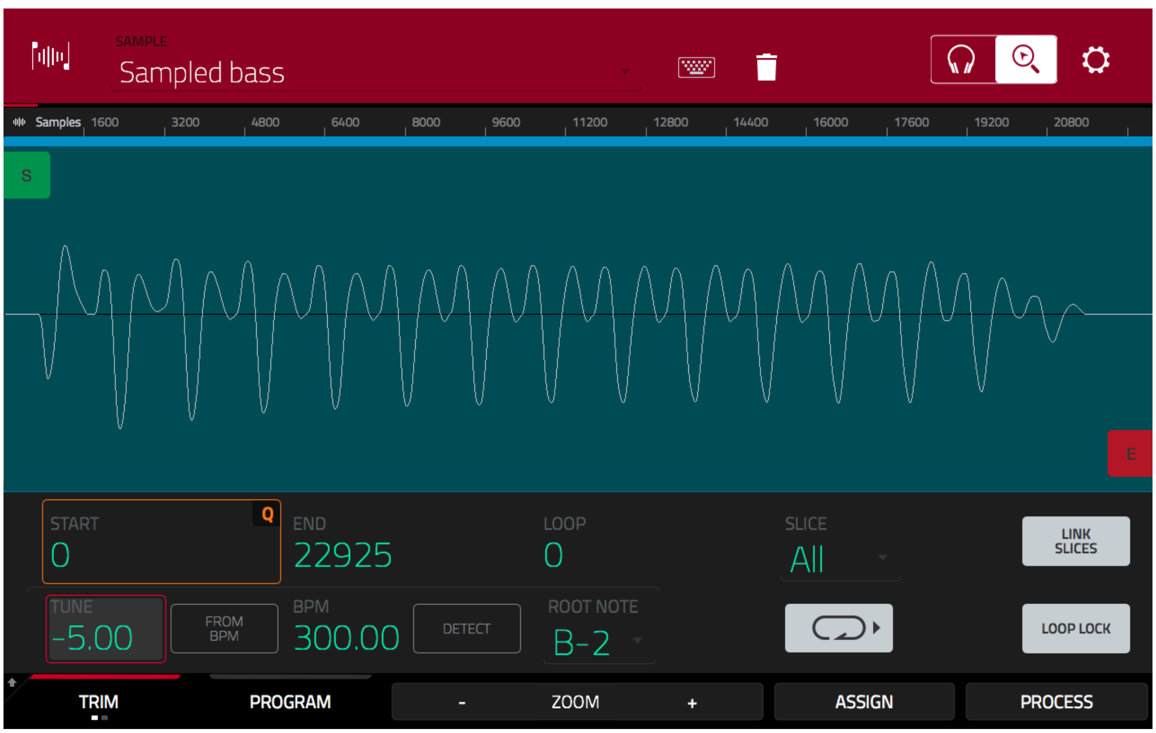
I also created an 8 bit version ('**not as nice snare**') so you can listen to the subtle differences. It's really just about how far you wish to take it – often in isolation, it's hard to hear the differences, but it often becomes more obvious once you begin incorporating these sounds into active projects.

Increasing Grit While Sampling From Vinyl

This classic technique uses the fact that when you tune a sample down, it creates a slightly darker sounding version of it (as we've seen when dealing with 'Tune' in program parameters). However tuning does change the *pitch* of the sample itself, which you may not wish to happen. This technique avoids this problem.

Firstly what you need is a record that naturally runs at 33 RPM. Find the section of audio you wish to sample and instead of sampling at the normal running speed, turn the turntable platter speed up to 45 RPM – record the sample at this higher speed using the methods described in the sampling chapter.

Press **KEEP** and open your newly recorded sample in **SAMPLE EDIT** and use the '**TUNE**' function to take the sample back down to its natural speed:



A **TUNE** setting of **-05.00** is required for conversion of 45 RPM to 33 RPM. Tuning down returns the sample to its normal pitch, but the act of tuning down adds a subtle amount of grittiness to the sample.

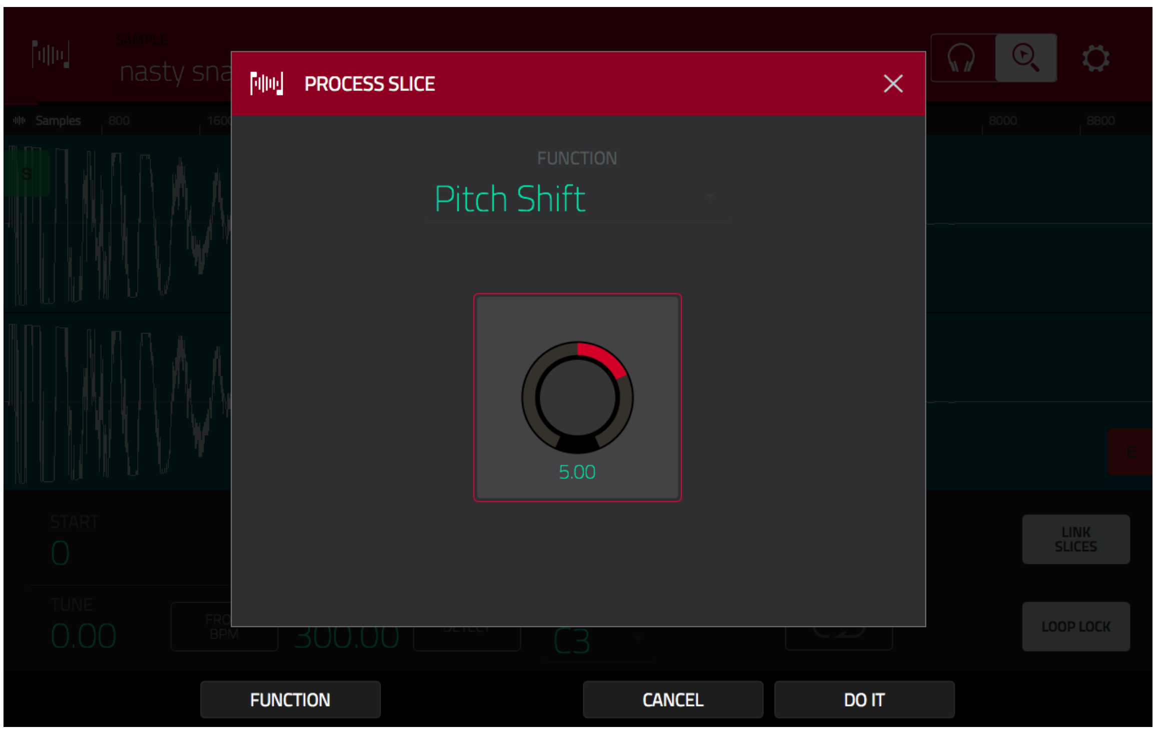
This method also saves memory, as your master sample is a lot shorter – probably not such a big deal these days as even the new standalone MPCs have GBs of storage and RAM to play with.

Using Sample Tune to add grit

If the previous sampling technique is not practical (i.e. you are working with an existing sample, or sampling from CD or 45 RPM records), you can still try out sample tuning in **SAMPLE EDIT**.

Select the 4 bit sample from the previous tutorial in this chapter - '**nasty snare'**. Enter a '**Tune'** value of **-5.00**. Things are much darker now, but as we expect, the pitch of the sample has changed. If this pitch change is undesirable, we can use a special sample process to help bring the pitch of the sample back up, while hopefully retaining the 'grittiness of the tuning-down process.

Go to **PROCESS > Pitch Shift**:



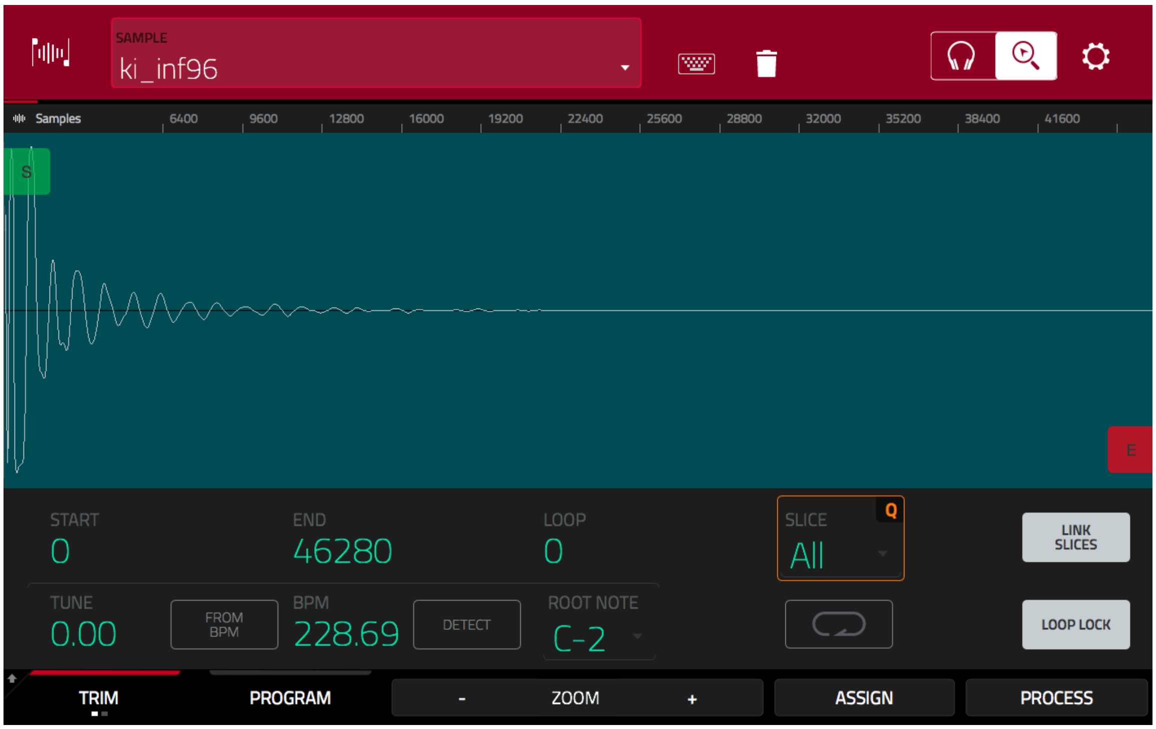
This process will keep your sample tuned down, but will use software algorithms to pitch your sample back up, without affecting the length of the sample and hopefully retaining the element of grit that initial tuning created.

Enter a **Pitch** value of **5.0** to counter the -5.0 tuning we performed previously, and hit **Do It.** You can load up my version if you prefer (**nasty snare pitch shift**) – they definitely have a slightly different sound to each other and the waveforms are very different.

The key to making grimy drums is to experiment with all the various techniques available to you; discovering new ways to create a unique sound is half the fun of sampling in the MPC!

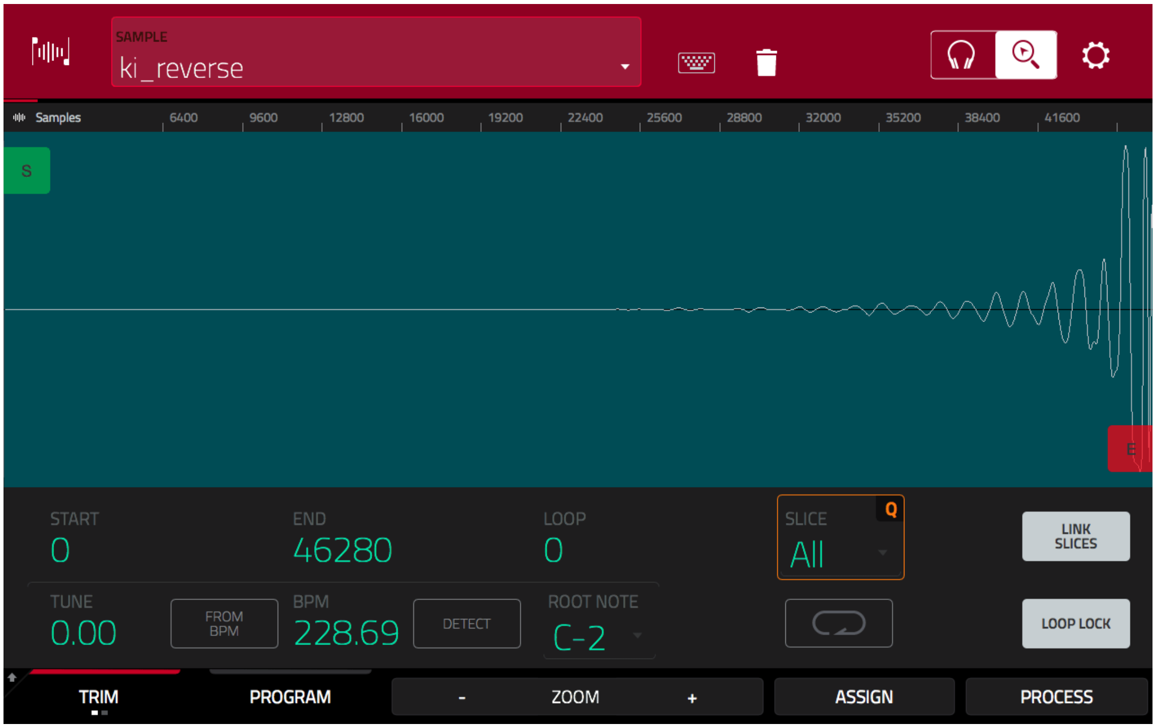
Emulating Scratching With 'Reverse'

We can do a pretty good job of emulating basis vinyl scratches in the MPC by using two different methods of reversing our samples. Load up the kick sample **ki_inf96** and select it in **SAMPLE EDIT**, it's just a standard, deep kick.

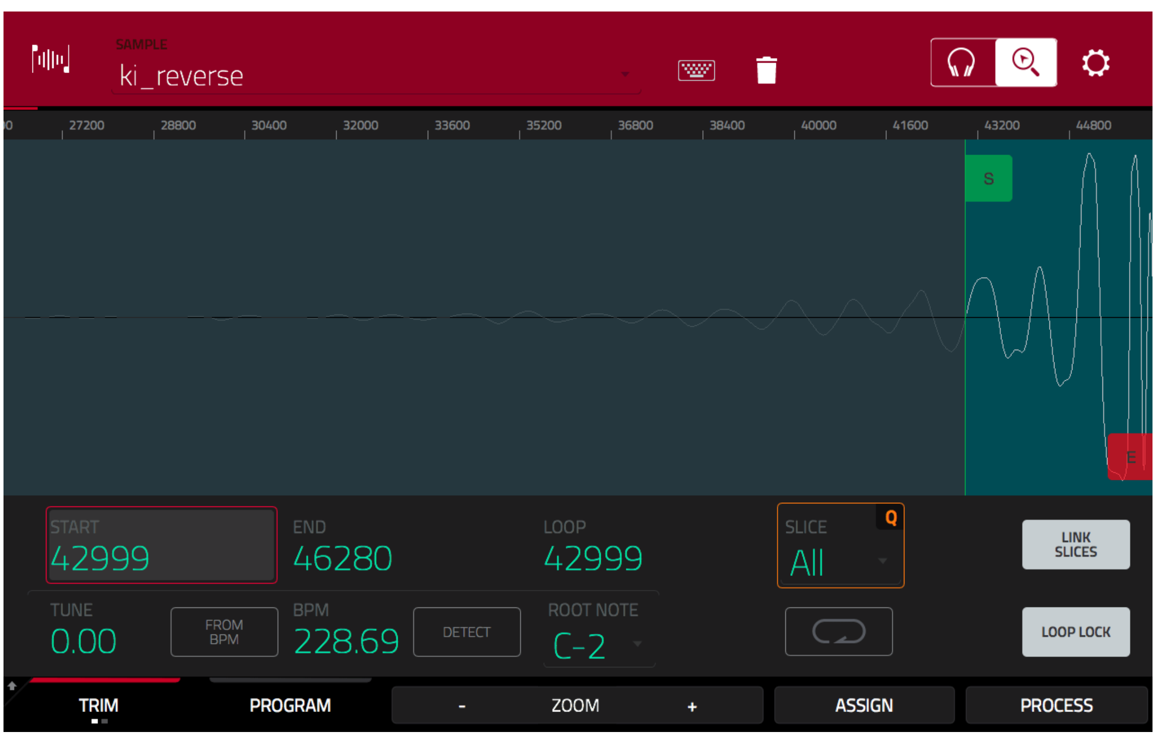


Select **PROCESS > Copy** and give the copied sound a name '**ki_reverse**'.

Now select this new **ki_reverse** sample and go to **PROCESS > Reverse**. Hit **DO IT** and press **pad 10** to preview the sound. As you probably guessed, **'Reverse'** simply reverses the portion of your sample between the start to the end points.

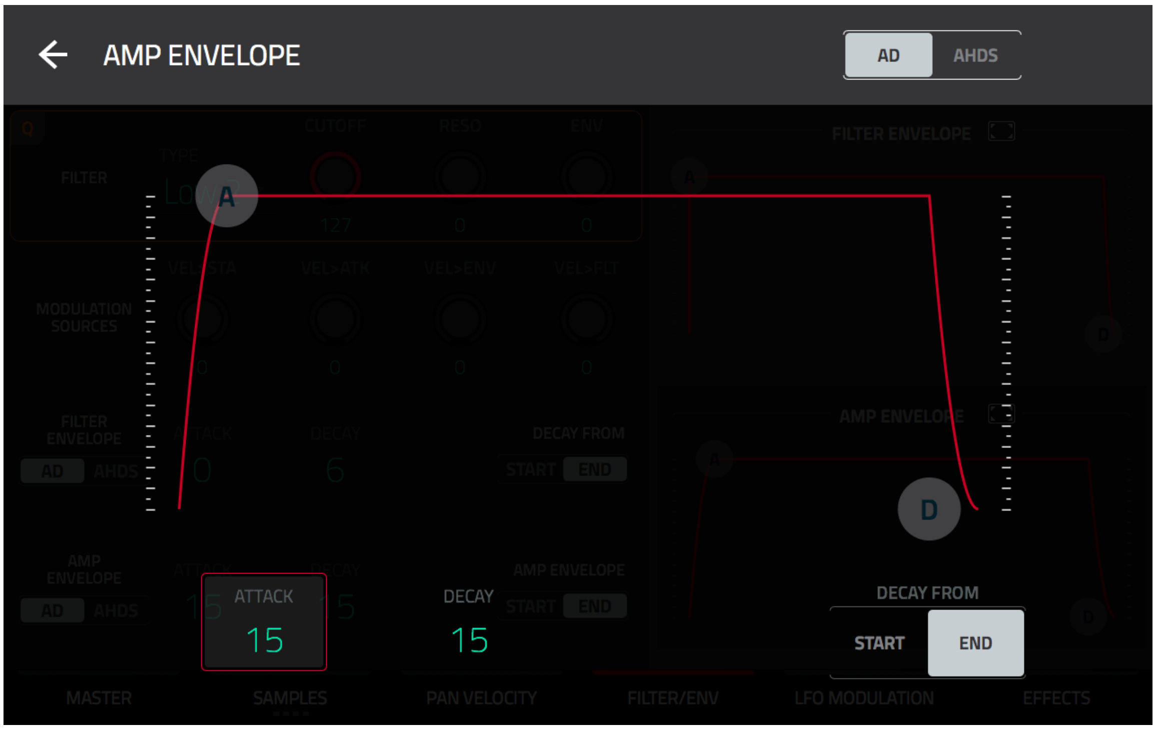


Go to **PROGRAM EDIT** and in **Program 001**, assign the original **ki_inf96** to pad **A01** and assign **ki_reverse** to pad **A02**. Play **A01** followed quickly by **A02**. So far nothing particularly realistic about this. However, head back to **SAMPLE EDIT**, select the **ki_reverse** sound and change its **start point** to approximately **42999** so a lot of that initial reverse build up is removed.

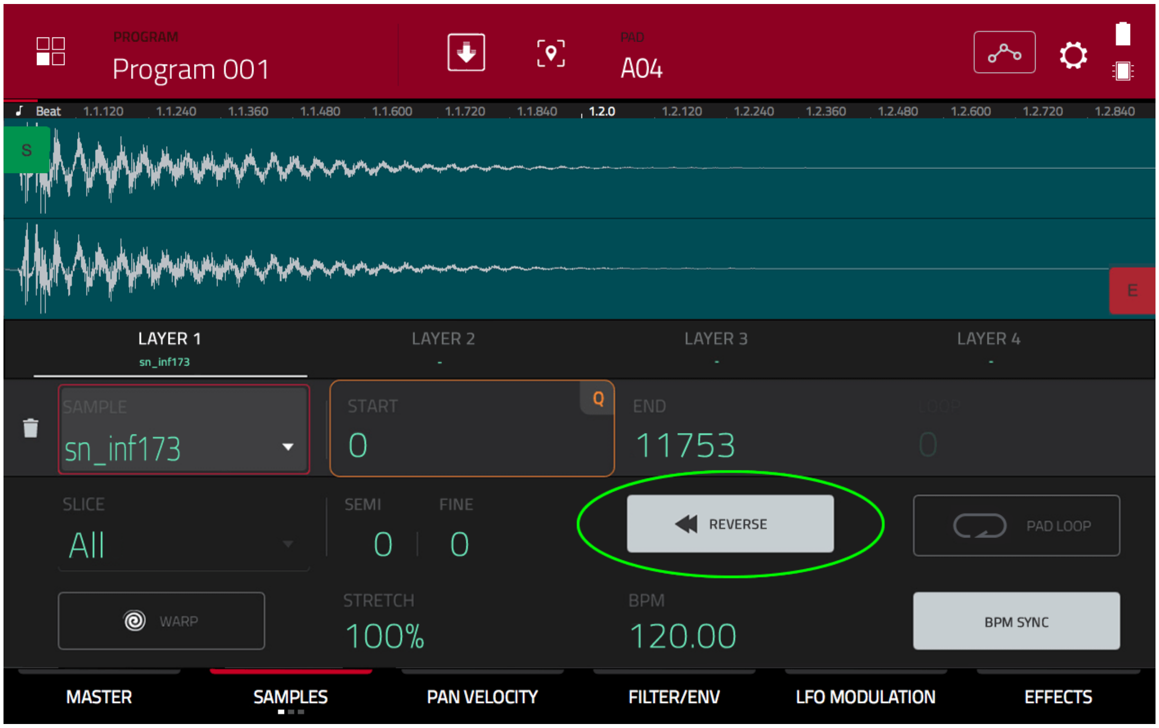


Now head back to **MAIN** or **PROGRAM EDIT** and play **A01**, followed quickly by **A02**, then **A01** again. You'll have to practice the timing a little, but that sounds like a pretty realistic classic kick scratch to me.

In **PROGRAM EDIT**, try increasing the **attack** value on to **A02** smooth out that initial reverse fade in, and maybe add a little **decay** to reduce the harsh click at the end – try setting **Attack** and **Decay** both to **15**.



We also have another, more efficient, 'reverse' option, which doesn't require us to make any copies of our sample. Load the **sn_inf72.wav** sample and assign it to both **A03** and **A04.** Select **A04** and go to **PROGRAM EDIT** > **SAMPLES**:

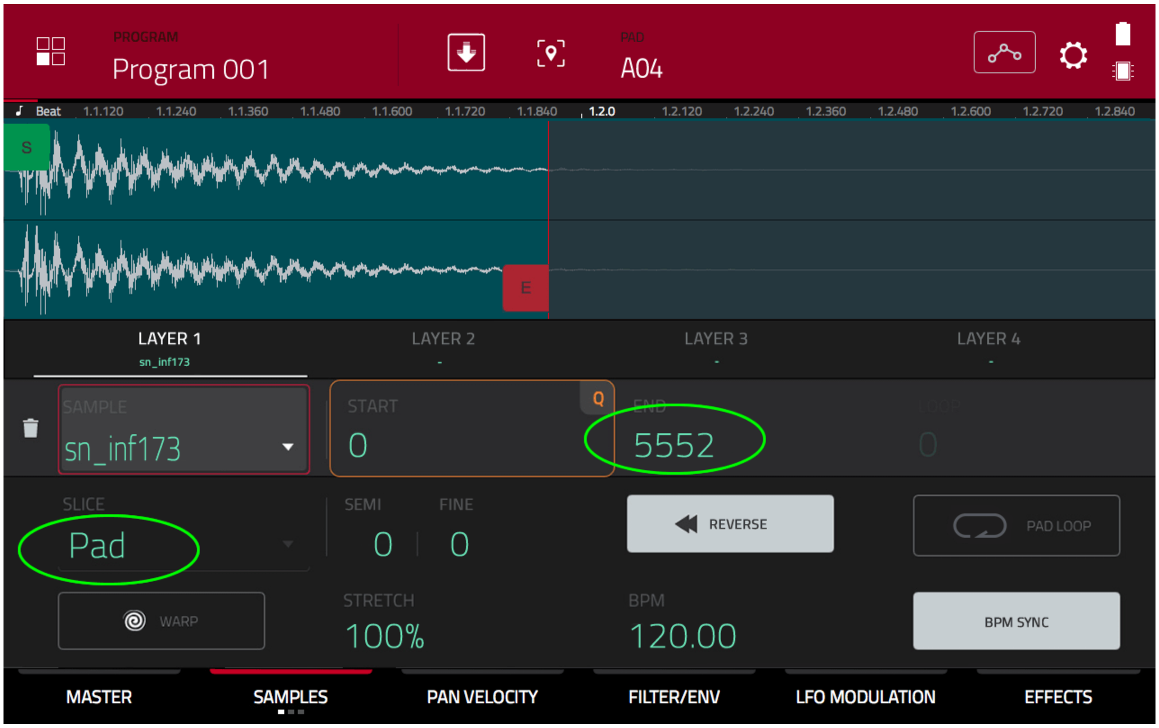


This time we're going to use Program Edit to reverse our sample; this can be done by activating the **REVERSE** button (as highlighted in green above). After doing this, preview pad **A04** to hear our reverse snare.

You can hear that the snare takes a while to come in and I want to hear it the moment I press the pad. So as we did last time, you could go into SAMPLE EDIT, make a copy and edit the copy to take some of the unnecessary space from it, *however* we can instead use a pad parameter to achieve the same result, once again avoiding the need to make a copy.

To do this we will utilise the **PAD START** and **PAD END** parameters that were introduced earlier in the book. As we've already seen, these effectively act like a unique set of start and end points for the sample exclusively within a sample layer itself, thus remaining completely independent of the 'master' sample. To activate the 'pad' parameters for this snare, set **SLICE** from **All** to **Pad**.

Now, as our reverse snare sound is taking its sweet time to come in, we want to reduce the amount of audio at the start of the file. However we must remember that our actual sound is being played in *reverse*, so in this unique example its actually the *end* of the original file that needs trimming, so in this instance it is the pad's **END** parameter that needs to be changed, rather than the START.



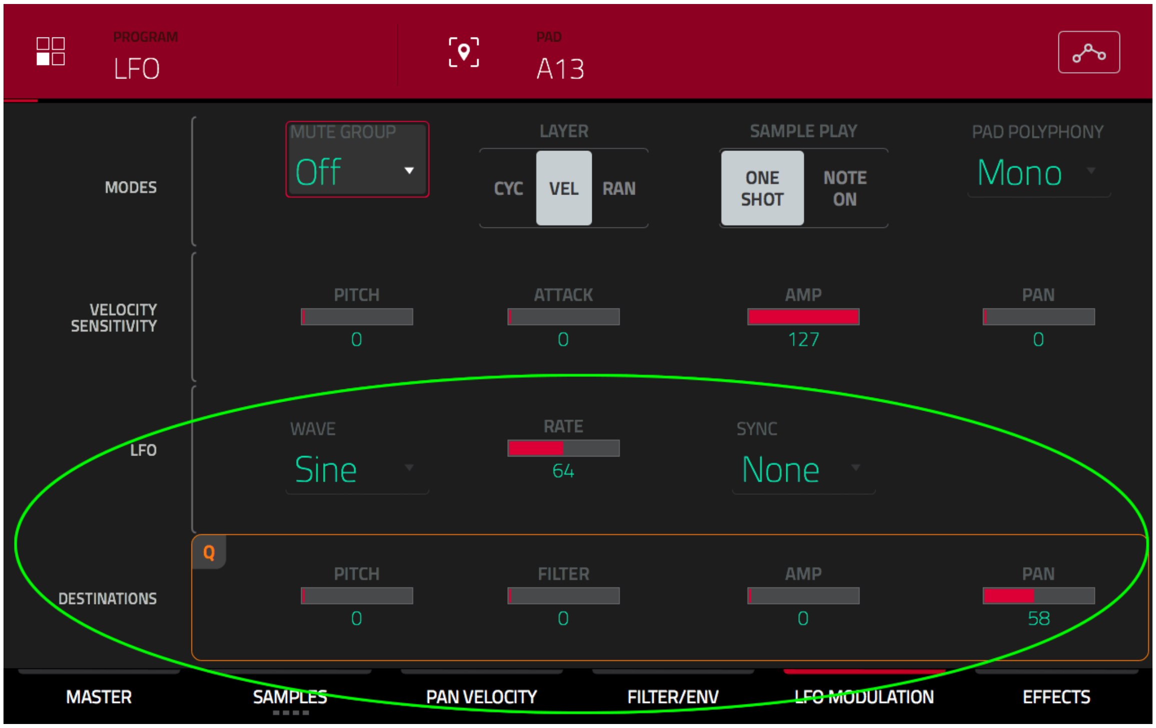
I set mine to **5552.** Now play between **A03** and **A04** to hear how the scratch effect works.

Load up the project file '**Fake Scratches.XPJ**, go to **MAIN** and play **sequence 1** (**Scratch Test**) – it's just a quick drum beat I laid down that uses the kick and snare scratches (the kicks on pads **A01**/**A02** and the snare on **A03**/**A04**).

Experimenting With LFOs

The MPC has built in **LFOs** (Low Frequency Oscillators) which allow some really interesting manipulation of the sounds assigned to a pad. An LFO itself operates at frequencies below the human hearing range, and can be used to *modulate* sounds in different ways based on the type of waveform they are oscillating at.

Locate the chapter 21 sub folder, **LFO,** load up the program **LFO.xpm**, go to **MAIN** and select **LFO** as the currently assigned program**.** Go to **PROGRAM EDIT** and touch the **LFO MODULATION** button. In this tutorial we'll be using the bottom two rows:

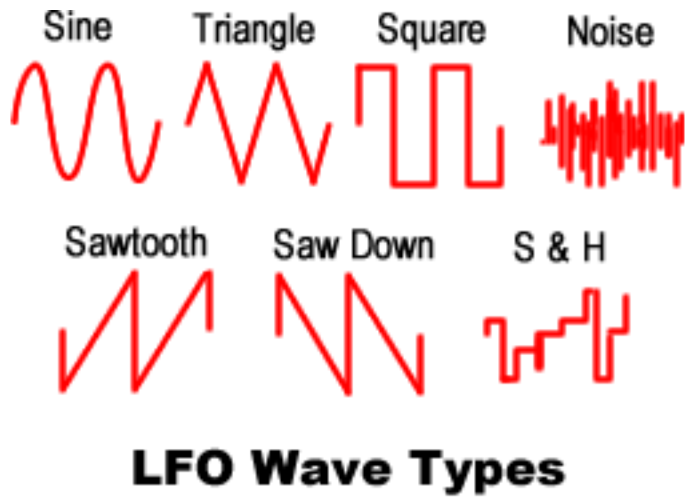


First, listen to pad **A13**. This is a sample of a nylon string guitar note. Nothing amazing there, but with our LFO we can manipulate this sound to emulate a couple of natural effects.

Tremolo Effect

Tremolo is quite a common effect used on guitar sounds – it gives your sounds a 'warbling' style effect.

First, observe the '**WAVE** parameter on the **LFO** row – this determines the shape of the LFO, which by default is a **Sine wave**:



Leave this set to **Sine**. Now set the **PITCH** parameter to a value of **32**. This setting will dictate the range of pitch modulation the LFO can control – basically, this will create variances in pitch (tuning) on your sample that follow the pattern of a sine wave.

Press **A13** to hear the effect so far. As you can hear, the tuning of the nylon guitar sound is pulsating high and low at a constant rate. Drop the **PITCH** setting down to **7** for a more subtle effect!

Now tremolo is usually much quicker than this, so let's speed it up. Increase the **RATE** to **81.** Preview **A13** again to hear the quicker tremolo effect.

Typically, tremolo does not stop at pitch modulation. The **AMP** parameter allows you to control the LFO's effect on the *volume* (amplitude) of your sound, and again this will modulate based on the waveform selected (currently a sine wave). Give this the maximum setting of **127** and preview pad **A13**. Sweet – again you could go for a more subtle effect by reducing this value to somewhere around **60** – it just depends on the type of tremolo effect you want to achieve.

Stick on some headphones and head over to the **PAN** setting. You've probably guessed that this allows the LFO to modulate the *panning* of the sample – set it to **58** to hear the warbling tremolo sound bounce between your left and right speakers.

Using the Sync Setting

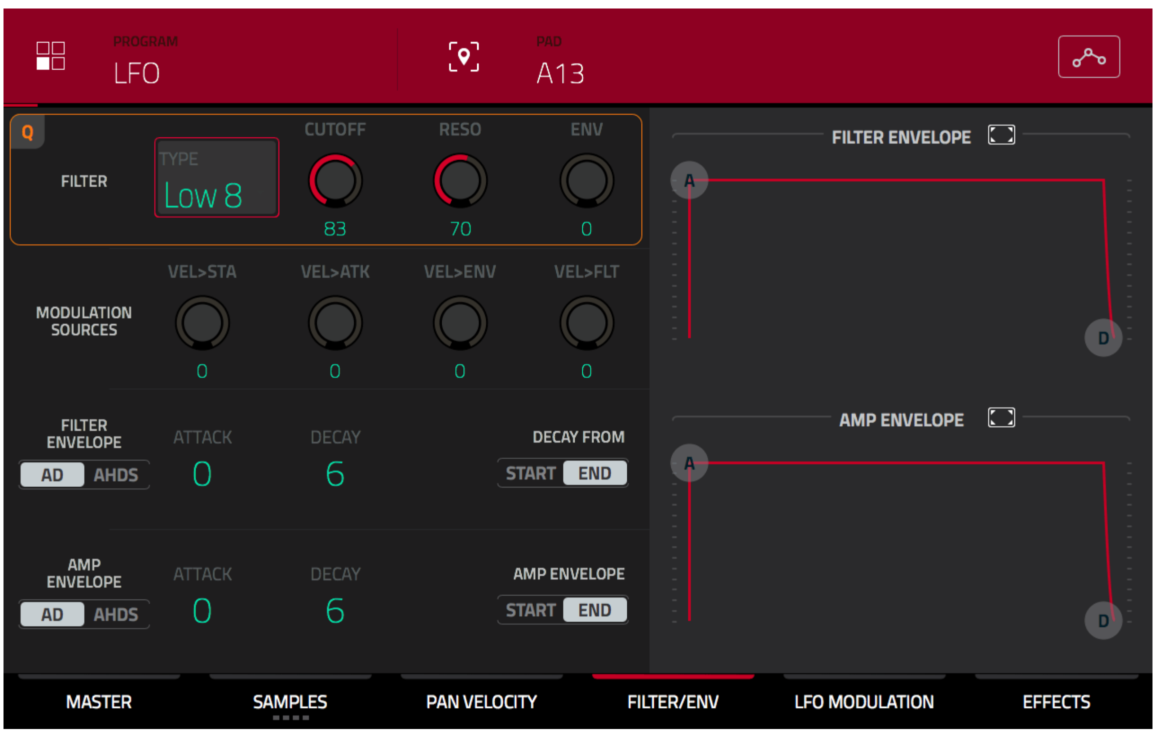
Instead of setting a rate value for your LFO, you can alternatively 'sync' the LFO to the current sequence tempo. Simply highlight the **SYNC** parameter and set it to **1/16** and the LFO will oscillate at **1/16ths** to the current sequence tempo of **120.0**. If you head over to **MAIN** and adjust the current sequence tempo and then re-trigger your pad, you'll hear the rate of modulation change accordingly.

Adding a Filter

You can really mess with your sound by getting the LFO to modulate a filter applied to the sound. On the **DESTINATIONS** row, set **FILTER** to the highest value of **127**. If you preview **A13**, you'll hear that nothing has changed.

Now press the **FILTER ENV** button to set up a bit of filtering. First, set a **TYPE** of **Low 8** (an 8 pole low pass filter) with a **CUTTOFF** of **83**. Preview pad **A13**.

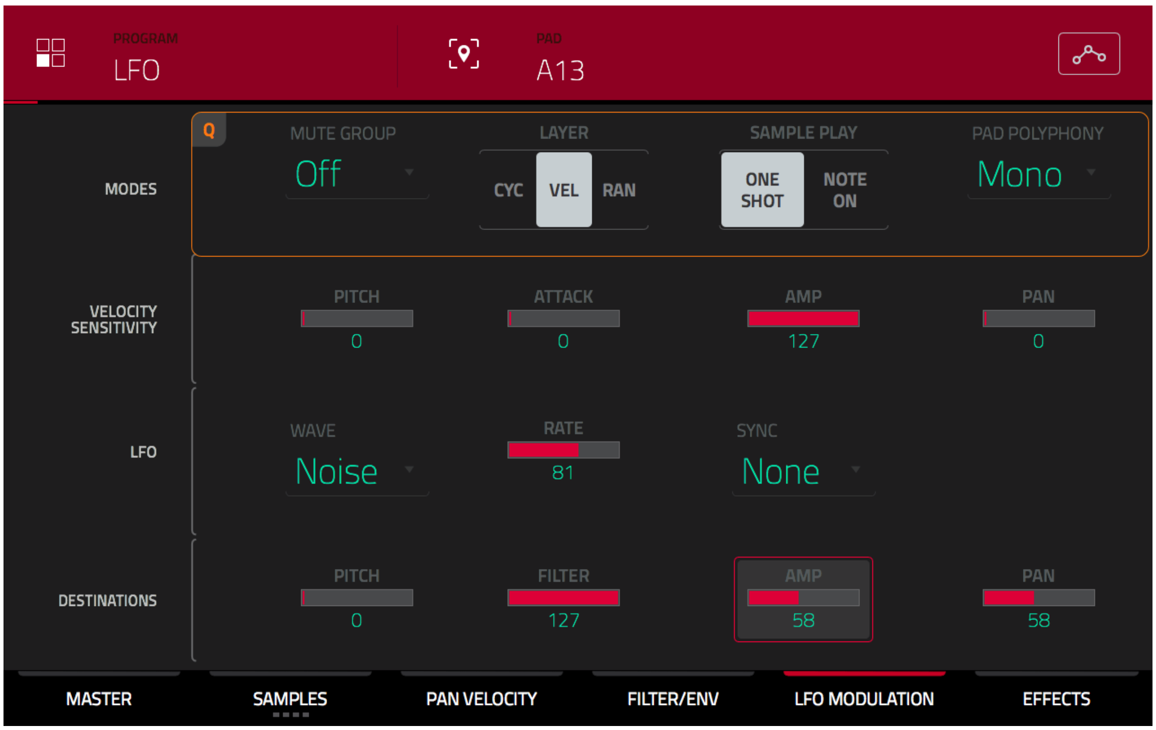
While A13 is playing, start increasing the **RESO** (resonance) value to **70** to give the filter a more shrill and piercing flavour.



Now start spinning through all the filter types to hear how they affect the sound – some have barely any effect at all, while others give some pretty crazy results. Try **BB 8P** and **Band 2** for example!

Emulating Vinyl Crackle

This one is a nice effect. In the **FILTER ENV** screen, set a filter **Type** to **Band 2** and set the **Cuttoff** at **127** and **Reso** at **70**. Now in the **Lfo Mod** screen select the **Noise** wave type, with all the other settings left the same as the previous examples (**RATE: 81**, **FILTER**: **127**, **AMP:58**, **PAN:58**).



Preview pad **A13** – instant vinyl crackle! Go back to the **Flt Env** screen and try playing with the **Reso** setting – the higher you set it, the more ‘modern’ the crackle sounds, while lower **Reso** value, the more ‘Vintage’ the crackle. This actually works with any sound assigned to a pad.

Take a listen to pads **A14**, **A15** and **A16** to hear my versions of these various experiments.

Wacky Bass

On pad **A09** is a bass sound – have a listen. I have copied this sound to pads **A05**, **A06**, **A07** and **A08** and have used different LFO wave types to create some very different results. Please refer back to the previous illustration showing the many different LFO waveform shapes the MPC offers.

Pad **A10** features a **square** wave and a **PITCH** of **50** – I’ve also set a BPM **SYNC** of **1/2T**. This gives us a short bass line riff which will ‘sync’ with whatever sequencer tempo you select.

Pad **A11** is a **triangle** shaped wave with a high **PITCH** setting (**109**) giving a weird pulsing effect. Pad **A12** is a similar effect, but sounds a little different thanks to the **Saw** wave used.

Pad **A05** uses '**Sample & Hold'** (S & H), which is a random stepped waveform. Each time you trigger this, it gives you a different result each time. **A06** uses a '**Saw Down**' (**SawD**) wave with **PITCH** and **AMP** settings applied to give a pulsating, pumping bass line.

A07 and **A08** both feature the **Noise** wave with differing **PITCH** and **AMP** settings to give different results. **A07** is gives the bass a grimy, distorted feel, while **A08** mimics an explosion.

Bit Crushed Drums

I've assigned a snare to pad **A01** which is pretty light and clean. Now check out pad **A02** – here I've applied a **Noise** LFO with a **PITCH** of **89** and an **AMP** of **73**. The result is a grittier sounding snare (the noise wave means the result will vary a little each time it is played).

Finally on **A03** I have a kick which I have 'grimed' up on pad **A04**. This time as well as the noise wave, I have set up a **Model1 filter** in the **FILTER ENV** sub screen. In the **LFO MODULATION** screen you can adjust the level of 'crackle' using the **AMP** parameter, where a drop in this will actually give more crackle to your kick. Reducing the **FILTER** setting will reduce the overall effect of the filter on the resulting sound.

There are limitless opportunities when using LFOs, you can literally play with the settings all day to manipulate the sounds on your pads.

Applying LFO to Individual Layers

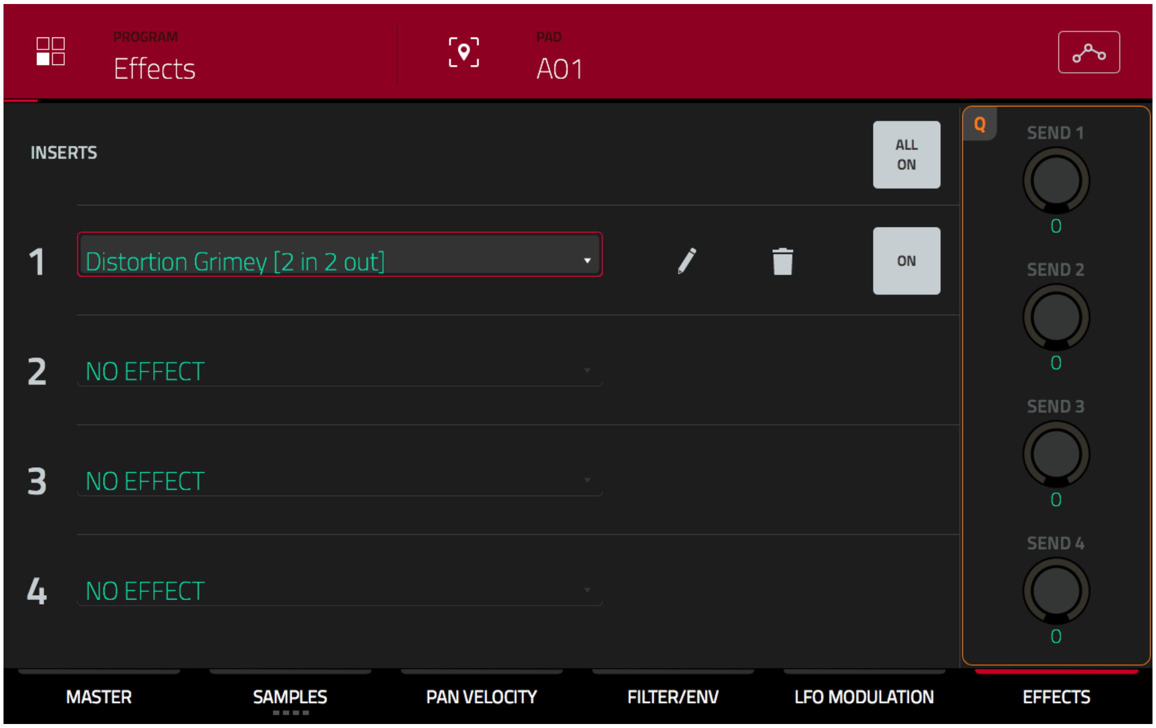
Remember that these LFO manipulations are applied to the *entire pad*, so when layering sounds, the LFO will affect all layers equally unless you adopt the '**Simultaneous**' method of layering we discussed earlier.

Effects to 'Dirty Up' Your Drums

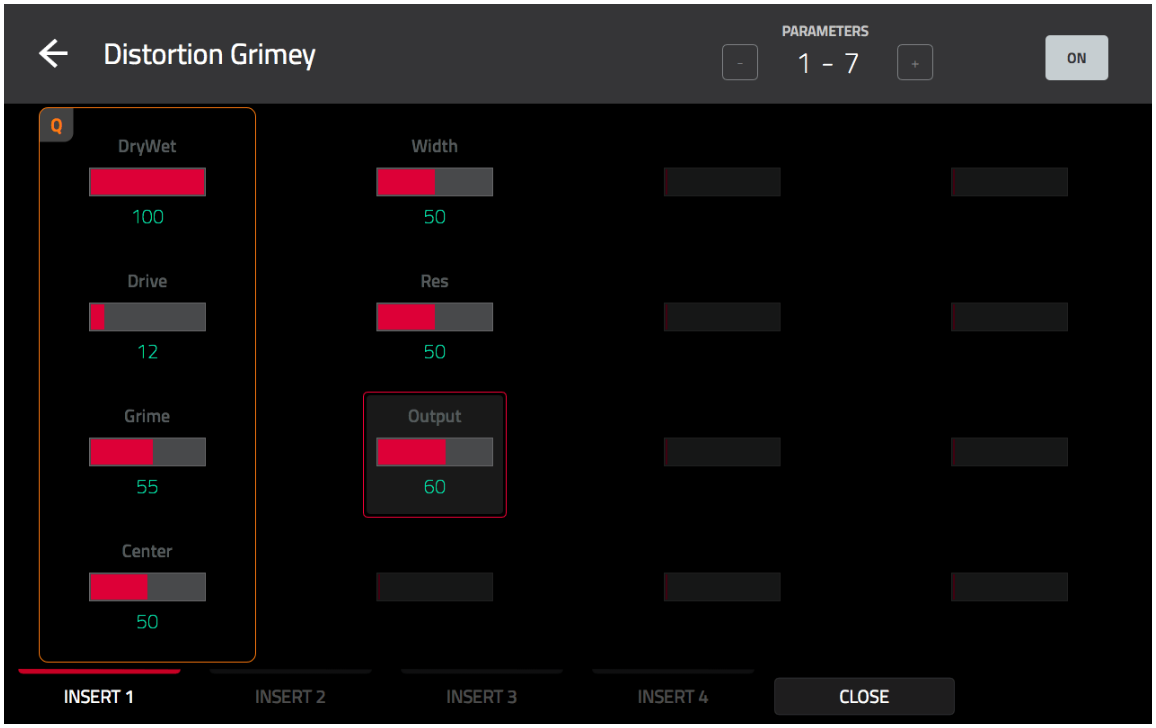
We can give drums some added grit using the internal effects. From the **Effects** subfolder, load up the program file **Effects.xpm**.

Distortion Effects

Select pad **A01**, and in **PROGRAM EDIT > EFFECTS** add the '**Distortion Grimey**' internal effect to **INSERT 1** for this pad:



To adjust the parameters for this effect, click on the **pencil icon**.



This is a pure 'Insert' effect, so ideally I will leave the **DryWet** parameter set to **100** (i.e. only the effected signal is heard). When you preview the snare with the default settings, it's quite noisy, so ideally we'd just tame this noise. The key setting for this particular effect is '**Drive'** – this dictates how hard the distortion is pushed. Try the above settings for some 'controlled filth'.

Notice how I've adjusted the **Output** to compensate for the small gain in output volume the effect was creating. My version is on pad **A02**.

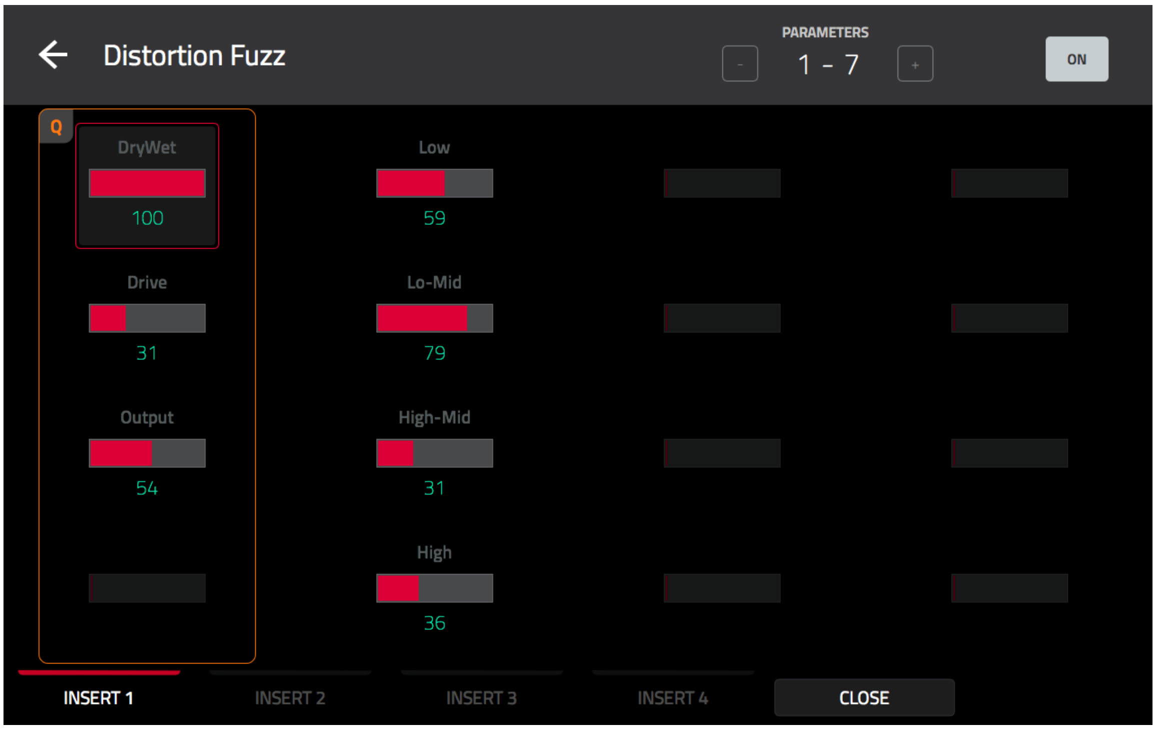
You can actually isolate a specific frequency band in the sound and apply distortion specifically to that band. This is controlled by the **Centre**, **Width** and **Res** settings, with **center** controlling the frequency being acted on (the number is, I assume, not an actual frequency value, but instead a relative value), **width** controlling whether the frequency affected is a narrow or wider band (i.e. a little like the Q value in an EQ), and **Reso** acting very much like the 'reso' setting in our filter section, with higher values producing more 'shrill' results.

Drop the '**Center'** value down to zero and begin to gradually increase it while previewing pad **A01**. As it increases, you can hear how the distortion is acting on the ever increasing frequency value. Stop when you hear a frequency that sounds good – I stopped at **18**.

Decrease the **Width** to help reduce the 'spread' of the frequency band – the smaller this gets, the less frequencies around the 'center' value are effected. I set mine to **29**.

Finally adjust **Reso** to taste – for a more 'natural' result, keep this low, otherwise use high settings for a more pronounced, piercing effect. I set mine to **18**.
My version is on pad **A03**.

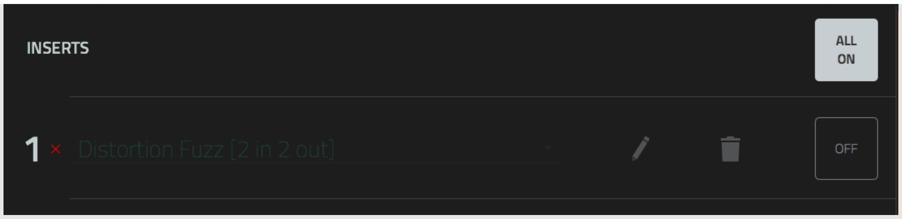
There are many other distortion effects, each one gives different results, with slightly differing settings. On pad **A04**, I have set up the exact same snare, but this time using the '**Distortion Fuzz**' effect, with the following settings:



This gives a nice, full bodied grit to the snare. I've used the dedicated EQ settings in the second column to shape the overall sound to taste. In my opinion, for adding some subtle crunch to your sounds, Distortion Fuzz is probably the one to go for.

Bypassing/Disabling Effects

To quickly disable an inserted effect, click the '**ON**' button next to the effect's name so it reads '**OFF**':



If you are using multiple effect inserts on a pad, you can turn off all effects by pressing the ALL ON button so it reads **'OFF'**.

Decimator

The Decimator is not the most natural sounding effect you can apply, it's very 'digital' sounding in my opinion, but still great for quite literally destroying a drum sound. On pad **A05** I have a different snare – select pad **A05**, turn '**INSERTS: ALL ON'** and select the '**Decimator'**. Hit the **pencil icon** to adjust the settings.

This is a simple plugin – for more 'decimation', simply increase the '**Decim'** value. Try a value of **60**. The **Bit.Reduc** reduces the bit rate of the sample – it's quite subtle compared to the effects of the Decimator setting, and cannot really be heard until you get to 8 bit or below. Try a setting of **8 bit** – my version is on pad **A06**. The result is a very 'grainy' sounding snare.

Resampler

While the Decimator 'decimates' in conjunction with adjusting the *bit rate* of your sample, the resampler also decimates, but this one works in combination with adjusting the *sample rate*.

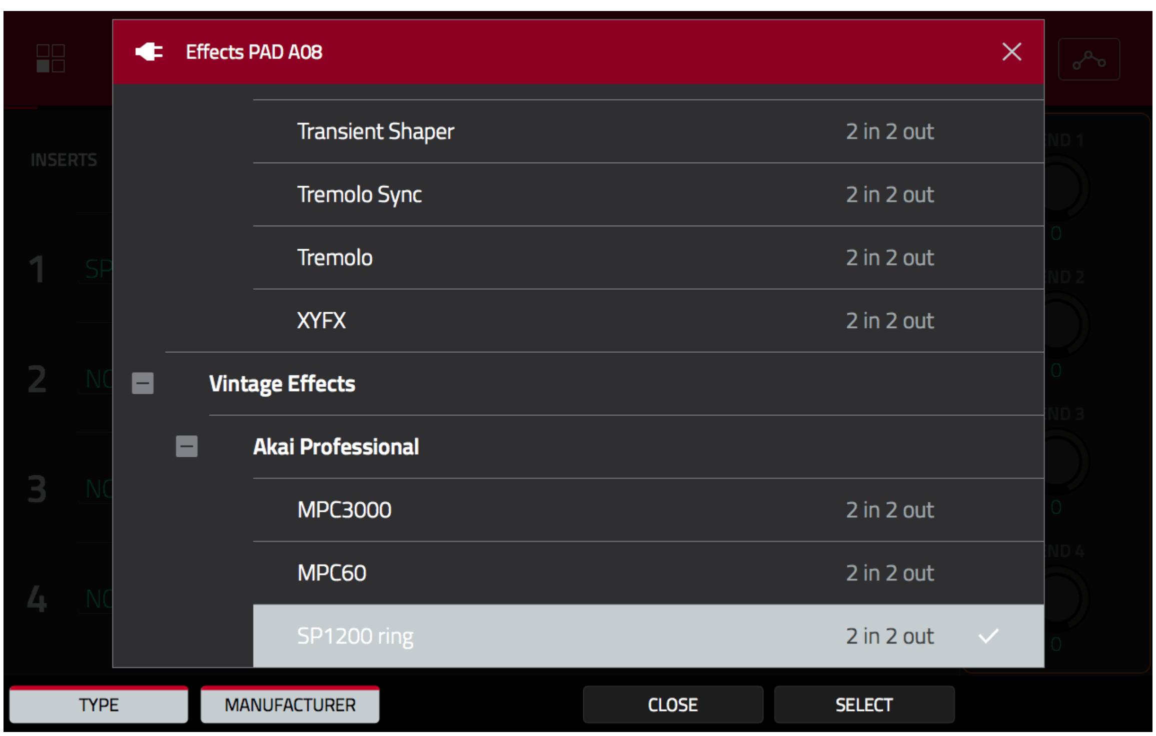
Replace the effect on pad **A05** to the **Resampler**. First, set a **Decim** value of **0** so we take the decimator out of the equation. Now, the lower the **Rate** (sample rate) value, the more dark and degraded your sound becomes, so start reducing that **Rate** value down to hear that change. I settled on **12**.

Now start increasing the **Decim** value and you'll start recognising the familiar decimator effect come into play. I prefer a lower value here, so try a **Decim** of **35.**

Listen to my version on pad **A07**.

Vintage Effects

Vintage plugins are a great way to add some warmth to selected pads. There are four vintage plugins, each one emulating the sound of a classic sampler; MPC3000, MPC60, EMU SP1200 and the more aggressive EMU SP1200 ring mode. They are accessed just like any other effect, and have been placed in their own folder:



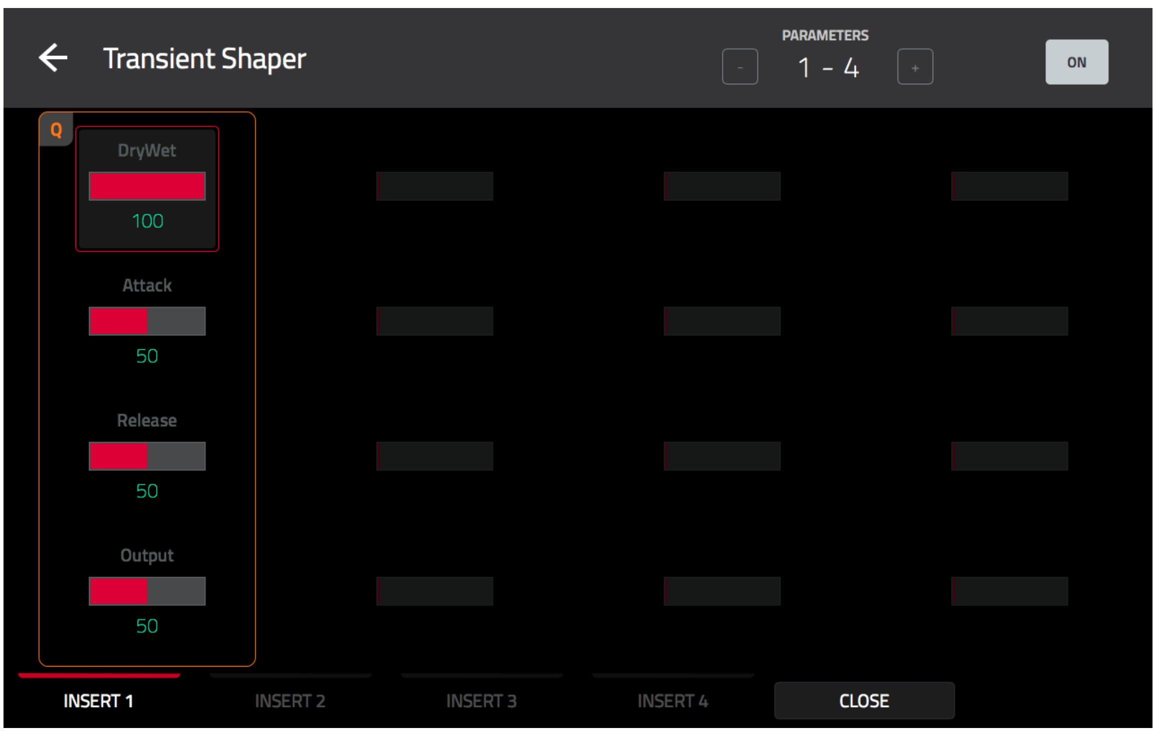
On pad **A08** I added the **SP-1200 ring** effect to **snare_layer_5** – this is the most obvious effect of all four vintage plugins, and as you can hear it grits up that snare nicely.

Try changing the insert effect to each of the other Vintage plugins and see if you can hear the difference. Also remember that you can place up to four effects on a pad, so try adding all four vintage plugins simultaneously!

Transient Shapers

A transient shape can dynamically 'mould' the shape of any sound, be it emphasising the attack or release portions, or controlling the output level of certain dynamic elements of the sound.

Select the kick on pad **A09** - set '**Inserts: ALL ON**' and for **INSERT 1**, select the **Transient Shaper** effect. Hit the **pencil icon**:



The transient shaper allows us to emphasise either the attack, or release portion of our sound. In the first instance, set the **Attack** and **Release** values both to 0. Now start to increase the **Attack** – while it may initially seem like the entire sound is getting louder, you are actually only increasing the level of the initial attack portion of the sound. Hence you have a kick with a much more aggressive attacking sound. This differs to the attack option in Amp Envelope, as that can only *decrease* the attack, not embellish it.

Set **Attack** back to **0** again and this time start increasing the **Release** value. At **70** it is clear that the transient shape has amplified only the release portion of your sound. Again, this differs to the decay or release found in Amp Envelope, as that option is really all about reducing and fading out this portion a sample, not amplifying it.

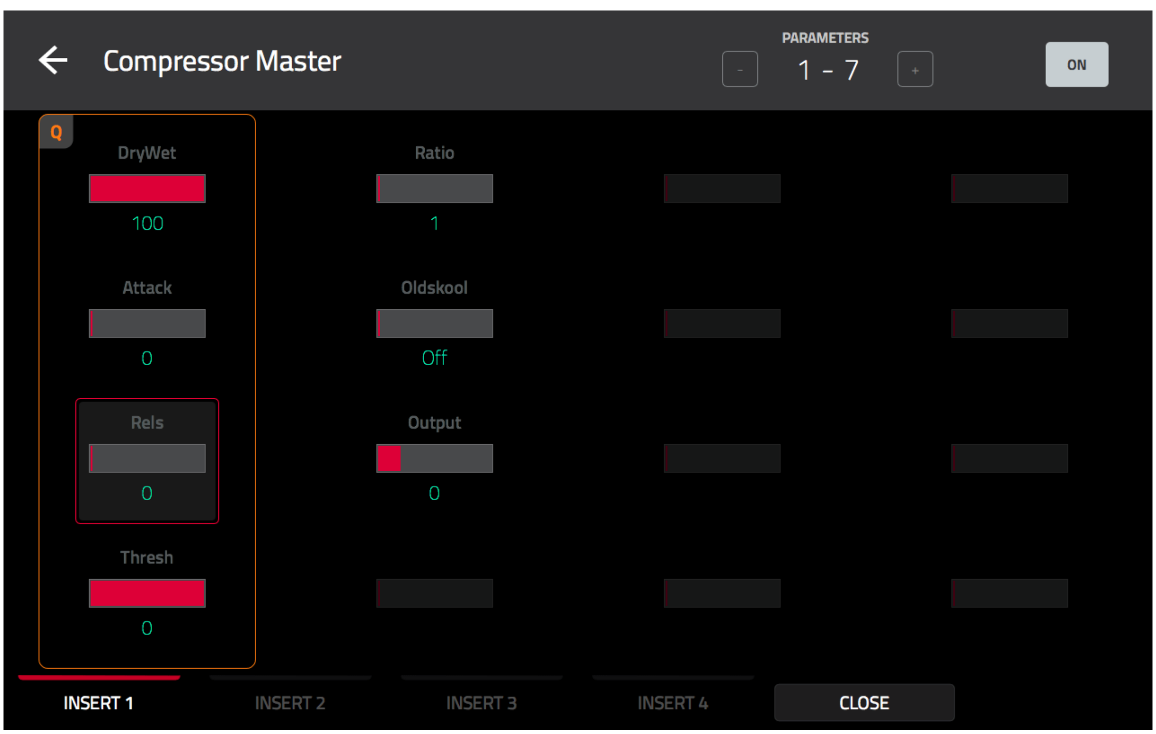
The transient shaper is a fantastically intuitive way to change the make up of any sound. On pads **A10**, **A11** and **A12**, I have included three examples of how the transient shaper can change the make up of your kick.

Remember to adjust the '**Output**' setting to compensate for any changes in output level this effect may have on your sound.

Compression

A compressor is another effect that can control the transients in a sound, and can perform a number of different tasks. A common technique is to 'compress' or 'squash' the higher peaks in your sound so that your entire sound has less variance between the loud and quiet parts.

Select the kick on pad **A13** – use this as a reference sound. Now select pad **A14**, where I have included the exact same kick. On **Insert 1**, select the **Compressor Master** and configure the following settings.



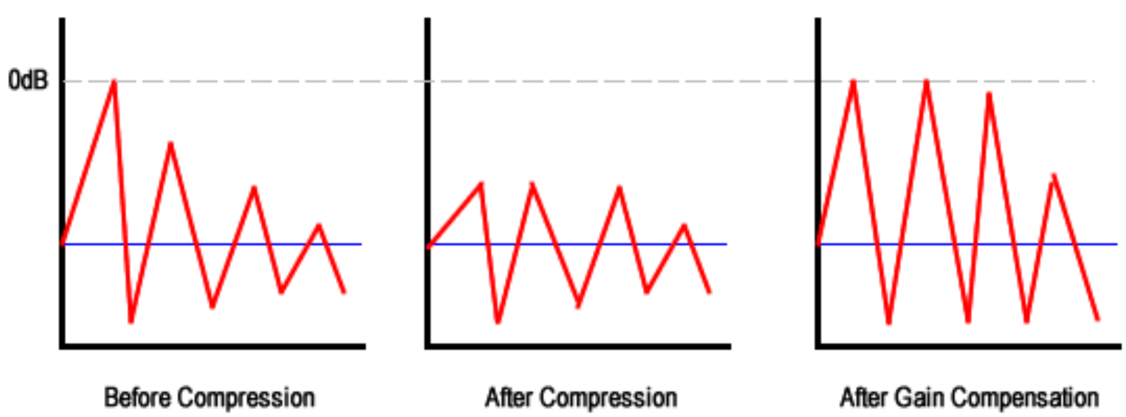
Basically, I have zero'd the compressor so currently it does nothing. Firstly increase the '**Ratio'** to the maximum value of **20**. This controls the amount of compression to be applied, this figure represents a compression ratio of 1:20.

However this has not actually changed anything to the outputted sound. This is because we have not yet set a **threshold** (**Thresh**). The threshold quite simply decides how loud your signal has to be before any compression is applied. Set a **threshold** of **-20**. Preview **A14** again and you'll hear the compressor now working.

You should hear that the signal has reduced in volume. This is because the compressor is *squeezing* down the loudest parts of your kick drum so the loud parts are closer in level to the quieter parts of the sample. To compensate we increase the **Output** setting so the output level of the compressed sample is the same as the original. To set this visually use your level meter; the physical meters in the MPC X or the software meter in the **PAD MIXER**.

An **Output** value of **14** should do the trick (it's slightly lower than A13, but an output level of 15 is a little too high). Now compare A13 to A14. It should be obvious that A14 sounds louder than A13. Why is this, when they both have the same output level?

Well, when you 'squeezed' the loud parts of your sound and then increased the output to compensate, you were simply bringing those loud parts back to their original output level. However when you did that, you were also increasing the output level of the quieter parts of the sound (the release) – these quieter parts were not compressed as much as the louder parts, as they would not have exceeded the threshold as much as the loud parts (they might not have exceeded it at all).



So now with these ‘quiet’ parts now much ‘louder’, the *perceived volume* of your sound is much louder. The output *level* is the same, but it simply *sounds* louder. The is why TV ads always appear louder than the actual programmes themselves – broadcasters are restricted by the output level they can broadcast at, but compressing the signal on adverts means they can make the adverts appear louder and hence get more attention.

So we have a louder sound on **A14**, but we also have a different sound. It’s more aggressive, with a very sharp attack and more pronounced release.

On pad **A15** I’ve increased the release to **100**. This controls how long the compressor will act and wait before applying itself again. Hit pad **A15** once to hear the effect. It’s reduced the release of the sound (it’s shorter and tighter). Now rapidly hit A15 multiple times and you’ll realise that at these fast speeds pad A15 is no longer being compressed. This is because the release value is so long that the compressor is not getting a chance to ‘act’ again, so it effectively remains ‘off’. You actually have to wait at least 1 second between pad hits for the compressor to act.

If you try this rapid hitting on pad **A14**, you’ll notice that the compressor ‘acts’ on every hit, simply because the release is set so short, so the compressor is able to do its thing every time.

This is obviously not something of concern for a single pad hit, but in a song where this sample is being hit in succession, it makes a big difference.

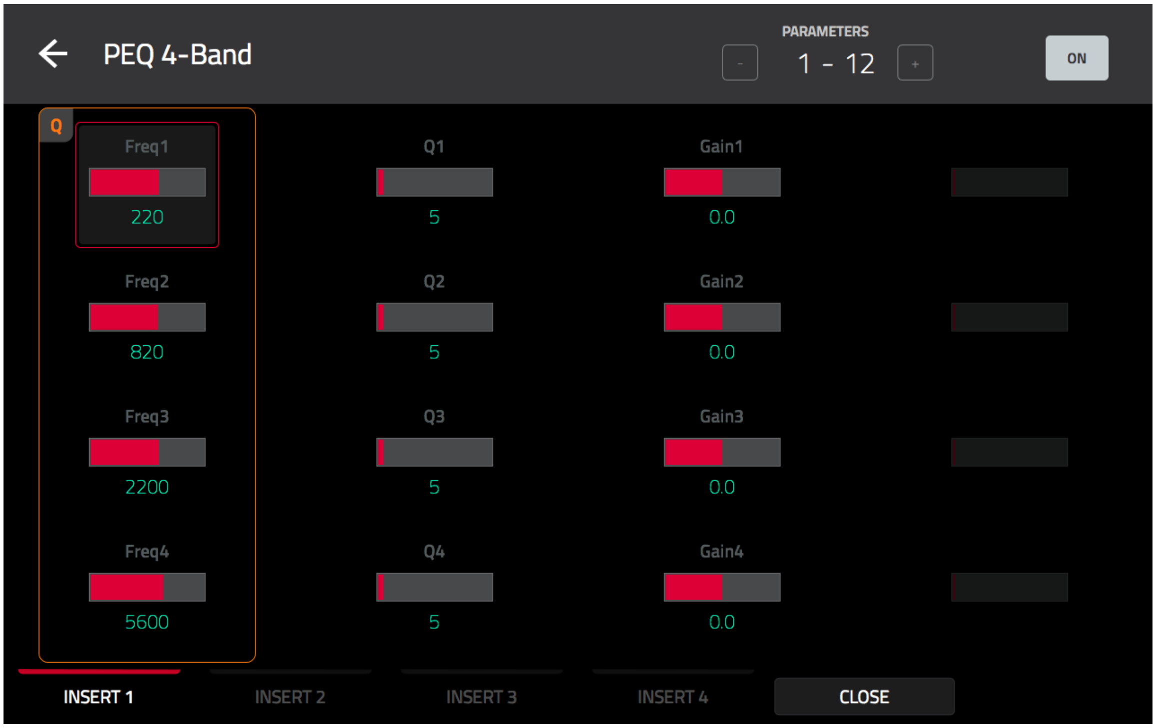
On pad **A16** I’ve set up a more gentle compression with a slower attack, higher threshold and less severe ratio. As you can imagine, the attack setting on a compressor dictates how long it takes for the compressor to act on the sound. The higher the setting, the longer the pause before the compressor acts. Pad A15 does not have that ‘clipping’ attack, and thanks to the lower ratio and threshold settings, has a less dramatic increase in perceived volume.

I’ve also engaged the ‘**Old School’** effect, which emulates the sound of a vintage compressor. Try switching this on and off to hear the subtle difference.

Using EQ in Sound Design

The EQ effects in the MPC are no different to using the EQ dials on a standard mixer and will allow you to select a frequency range and cut or boost this range to change the sound of your pad, be it in a creative or a purely functional way, e.g. removing hum from a mic recording, removing hiss from an old vinyl sample, or giving a little boost to the low end of a kick drum.

Select the clap on pad **B01** where I’ve already added a **PEQ 4-Band** effect to **Insert 1** with the default settings:



This is a fairly basic 4 band EQ, and it allows you to boost or cut up to four specific frequencies from your sound. The first column lets you specify which frequencies should be targeted. The last column allows you to set the **gain** for this specific frequency – a negative number is a cut, a positive number is a boost.

The central column controls the **Q** value. This dictates the bandwidth or 'spread' of frequencies that are affected around the specific frequency you've set – a low Q value will give you a *larger* bandwidth. So for example if you have set a 'Freq' of 200 Hz, a high Q value will mean that any gain changes you make will generally only make changes to the 200Hz area of your sound. But set a lower Q and you'll find many frequencies either side of 200 Hz will also be affected.

A common EQ task is to apply a small boost to lower frequencies, giving your sound more low end presence. Set a **Freq1** of **150**, leave **Q** at **5** and set **Gain1** to **9.0**. Preview the clap now and you'll hear it has more low end. You can now experiment with the Freq1 settings to find a more 'sweet' spot for this sound. I found the **100Hz** gave me a more natural sounding bass boost. My version is on pad **B02** – if you compare the unaffected version, you can hear how that boost has subtly changed the low end of the sample.

Generally speaking though, you should use EQ boosts with care and it is often better to *cut* a frequency rather than boost – it can often achieve a similar result, but in a more natural way. Check out **B03** – here I have actually performed an aggressive cut of **Freq4** at **9884** Hz which has removed a lot of top end from the sample, giving a darker clap, more 'bassy' feel, without the artificial boost at the low end (which can sometimes muddy a mix).

If there is a specific character of the sound that you simply do not like, EQ can be used to identify and reduce or pretty much remove this frequency from your sound.

The first step is to have a guess where the problem frequency may be. For example, let's say you don't quite like the 'crack' in this clap – just like snares, the main body and crack tends to fall in the middle frequency regions, between **800-2000** Hz. Let's use **Freq3**. Initially, set a boosting gain of **18.0** so that any frequency we select is heavily boosted.

Now select a more narrow **Q** of **75**. This means we will be focusing our frequency search to a more specific area (you can set an even higher Q if you wish).

Now using **Q5**, turn through the range of frequencies from around **500** and upwards while continually previewing your sound. Listen out until you hear a frequency that seems to represent the bulk of your clap 'crack'. I stopped on **1678 Hz**.

Now simply set the Gain to **-6.00** to remove a lot of this frequency. The more cut you perform, the less natural it will sound, so typically it's really just about reducing it in a more subtle way. You can try it at -18, but it sounds really artificial. Listen to my version on pad **B04**.

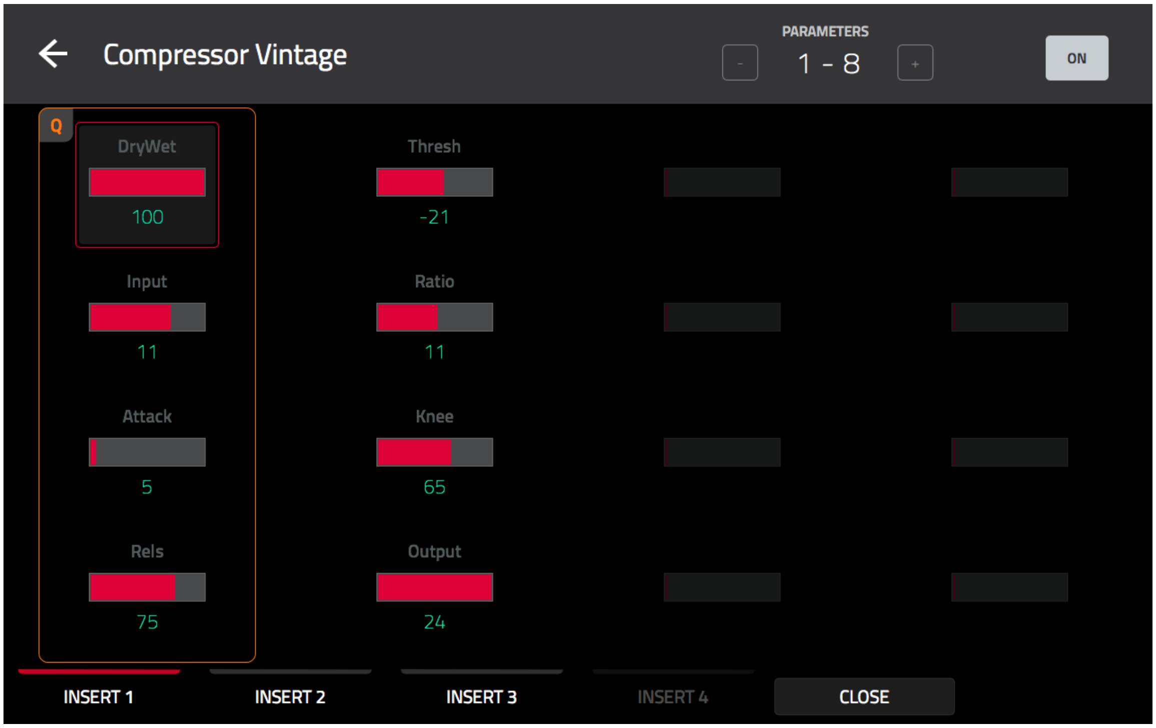
Creating a 'Live' Sounding Snare

Reverb and delay are usually considered 'send' effects and are typically *mixed* with the existing sound rather than completely *replacing* the sound like an insert effect does.

However there is no rule that says you cannot use reverb or delay as an insert effect, so let's look at a way we can add some subtle ambience to a snare in conjunction with some compression and EQ.

On pad **B05** I have placed a snare which was extracted from a close mic 'stem' from a multitrack drum recording. Typically when recording a snare you would also mix together the signal from an overhead microphone which would result in a more open snare sound with some of the natural room ambience. Without this overhead mic mix, we're left with quite a flat and short snare sound, but we can try to emulate the more natural snare sound using a combination of effects, as well as take the opportunity to apply a little creative EQ. Take a look and listen at pad **B06** to hear the result.

On pad **B06** I have inserted a **Compressor Vintage**.



I have driven it reasonably hard (the '**Input**' value), with a fairly low threshold and medium compression ratio. This really helps 'open up' the snare. I've also used the compressor to help boost the rather quiet volume of the original snare – you can hear it sounds much louder thanks to the initial 'squeezing' and subsequent output compensation applied.

On **Insert 2** I have applied some '**reverb small'**, with very gentle settings that represent a short, subtle room ambience. Notice that unlike the insert effects, reverb is a send effect, so I have adjusted the **DryWet** value to control the mix of effected signal with the unaffected. I've used a **DryWet** value of **33**, i.e. 33% effected signal.

I've then used a **2 Band EQ** on insert 3 to apply some boost in the 5600 Hz region, just to give some additional brightness to the snare.

Finally if you take a quick peak in the **FILTER ENVELOPE** sub-screen, the original sound has a bit of noise at the end which was being amplified by the compression, so I've added a little **decay** to fade this out a little.

All in all, a much more lively snare!

Using reverbs and delays in sound design

If you are using programs for sound design purposes to create ‘standalone’ samples it’s not considered good practice to add reverbs or delays because the ambience it adds is set in stone’ and could make the sound unusable at a later date.

This is of course not a *rule*, and there is nothing stopping you using a reverb or delay as an insert effect if you wish, especially if you feel it’s adding something unique to the sound. In the example above, adding the reverb very subtly has given the snare a very slight ambience that isn’t actually identifiable, it just gives the snare a bit more life, so I would be more than happy to use this, even if the sound was destined for ‘standalone’ use in a separate drum kit.

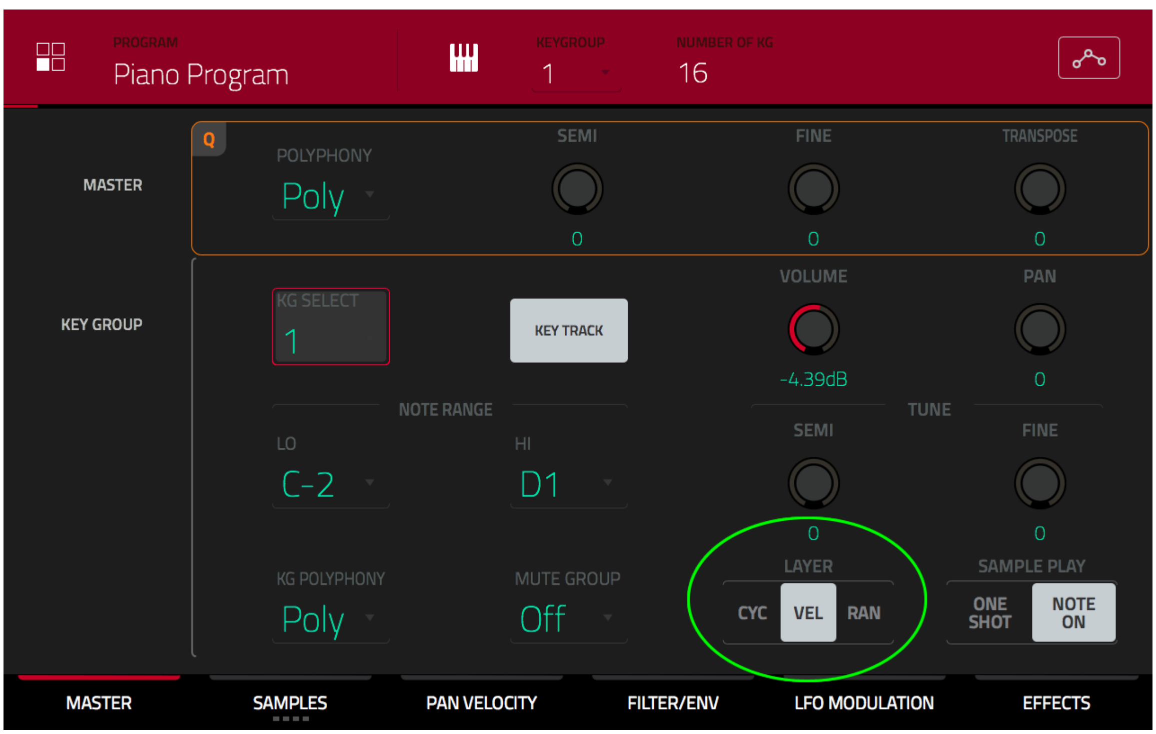
022 Advanced Keygroup Editing

We first discovered KEYGROUP programs back in chapter 010 where we created a simple multisampled piano instrument. In that chapter I want to continue using that same program to introduce some more advanced keygroup editing concepts.

Timbre Emulation Within a Keygroup

From the **chapter 022** folder, load up the basic keygroup program we made at the end of **chapter 010**, **Piano Program.xpm**.

Now, just like a pad in a DRUM program, each keygroup in a KEYGROUP program can hold up to four samples, with each sample being assigned to its own layer. The 'layer play' options for a KEYGROUP program are in the '**MASTER**' screen:



The default layer play is '**VEL**' (Velocity) just like DRUM programs, so with this current setting your layers can be set to either play together or can be velocity switched (by using velocity ranges in the **PAN VELOCITY** screen). So for example if you had recordings of your C1 piano note at hard, medium and soft velocities, you could assign these to layers 1 to 3 and set up your velocity switching ranges accordingly, just like we did with the multisampled drums in chapter 017.

But what if you don't have these additional 'dynamic level' samples, what can we do to bring some timbre variation? Well we'll see later that we still have some velocity sensitive emulation options available in keygroup programs, but first let's look at a way in which we can cleverly use our existing samples in combination with random play and the concept of *overlapping* keygroups.

Utilising the Root Note setting

First, remember how all our samples have their **root note** already set in SAMPLE EDIT? This means that our keygroup program is well aware of the true musical pitch of the sample - you can in fact put it anywhere in any keygroup and the MPC will adjust its tuning to play at the correct note value.

So there is nothing stopping us, for example, placing the **piano-C1** from keygroup 1 into keygroup 2 – the MPC would simply think *'Hmm, we have a C1 note here but it has to play back in a keygroup covering Eb1 to F1, no problem I'll just tune it up a few more semitones!*'. The MPC would then tune the sample automatically for you, all thanks to the root note setting.

This means we can place multiple piano notes on different layers in a keygroup and combine them with 'random' layer play to produce some realistic timbre changes each time we hit a pad. Let's explain this all with a practical example.

Take a look at **keygroup 3** – this is set to play the notes **F#1** to **A1** using **piano-G1** on **layer 1**. To hear this keygroup just play pads **C11** to **C14**. Now just temporarily change the sample assigned to layer 1 from **piano_G1** to **piano-C2**. Now play pads **C11** to **C14** again. What's changed? Well those pads are still playing the correct notes, because the **piano-C2** sample is identifying its true pitch to the MPC via the ROOT NOTE data set inside it.

Obviously tuning a C2 down to F#1 (the 'Low note' of this particular keygroup) means we are tuning down by up to *six semitones*, so you may notice there's some degradation of sound. The alternative is to go in the other direction and replace the **piano-C2** sound with the **piano-E1** sound – try it. Once again, that magic root note setting means all the notes in this keygroup remain in the correct musical key. This time we are tuning the **E1** to the '**High Note'** of **A1**, which is a tune up of *5 semi tones*. Not only is this a smaller number of semitones, but a tune up tends to degrade the quality of the original sound much less than a tune down.

However there are no rules here – just use your ears and decide what sounds acceptable. In my opinion, I feel all of these sound fine so let's actually set all three samples up in this keygroup program! So do the following in the **SAMPLES** screen:

- Set layer 1 back to piano-G1
- Set layer 2 to piano-E1
- Set layer 3 to piano-C2

Now in the **MASTER** screen, change **LAYER** to either **RAN** (random) or **CYC** (cycle). Now when you play any of the notes in this keygroup, you'll get the MPC playing back any one of these layers. Prefer one sample over others? Give it priority by assigning it again, but his time to **layer 4**, so you have a 50% chance of this sample playing compared to a 25% chance for all other samples (for this higher priority layer, you'll probably want to assign the 'correct' note, **piano -G1**). Here's what it looks like in **SAMPLES screen 2**.

Try one of the pads in the current keygroup range – i.e. pad **C12** – keep hitting it to hear the layers randomly playing back, all at the same pitch, but with different timbre.

So yes this method requires some sounds to be tuned more than we may normally prefer, but unless we are dealing with very delicate acoustic sounds, I feel we'll get away with it and simultaneously reap the benefits.

So now repeat this process over all 16 keygroups. Remember as all your keygroups are now being set to random play, you can first select '**ALL**' keygroups and set them all to random play simultaneously. Then select a specific keygroup number and set up the appropriate sample assignments, simply taking the sounds either side of the ‘primary’ sound that’s already assigned – so for example, if you have **piano_E4** assigned to a keygroup, **layers 2** and **3** should be **piano_C4** and **piano_G4** respectively, followed up with **piano_E4** again on layer **4**.

You can load up **Random Timbre.xpm** to hear my version of this program.

Tuning & Transposing Your Keygroup Program

In the **MASTER** screen you are able to change the global tuning of your program. If you wish to make some small tweaks to the tuning of your keygroups, typically because *all* your notes are slightly off from ‘concert pitch’, use the **Semi** and **Fine** parameters (however if only some individual notes are slightly out of tune, use the ‘**Tune**’ setting for the affected samples, in **SAMPLE EDIT**).

You can also ‘**transpose**’ your entire keygroup program using the **TRANSPOSE** parameter. This can be useful if you’ve previously recorded a performance to your sequencer in the key of C, but later find you need it to actually be in the key of D. Rather than editing your recorded MIDI notes in your sequencer, simply ‘transpose’ your entire keygroup program up **2 semitones** (a **TRANSPOSE** setting of **2**).

Transpose can also be used to change the position of the lowest ‘usable’ note in a keygroup program. For example, as we’ve already seen, bank A often contains notes far too low to use with acoustic instruments, so you could ‘transpose’ your entire keygroup program 2 entire octaves (**TRANSPOSE**: **24**) – it would still be a C note on pad **A01**, but rather than a C-2, it would be a **C0**, which generally is a much more useful note to start an acoustic program with.

‘Pseudo’ Stereo

One way to save memory in a keygroup instrument is to use mono samples instead of stereo, which is what we’ve been using throughout these keygroup tutorials. However many instruments do sound much better in stereo; other than sampling the original instrument in stereo, an emulation of stereo can be achieved on a mono program by applying a stereo effect across the program (e.g. a reverb or delay), or we can utilise panning across two layers in each keygroup, with one layer panned hard left, the other hard right.

Rather than pan the same sample left and right we achieve a fantastic emulation of stereo by using root notes. Load up the program '**Psuedo Stereo Piano.xpm**', go to **PROGRAM EDIT > PAN VELOCITY** and select any keygroup:

Here you can see that for each keygroup I have assigned two samples; on layer 1 is the 'proper' sample for this key range, while layer 2 features the nearest neighbour, which as we've seen will automatically tune itself to the correct pitch.

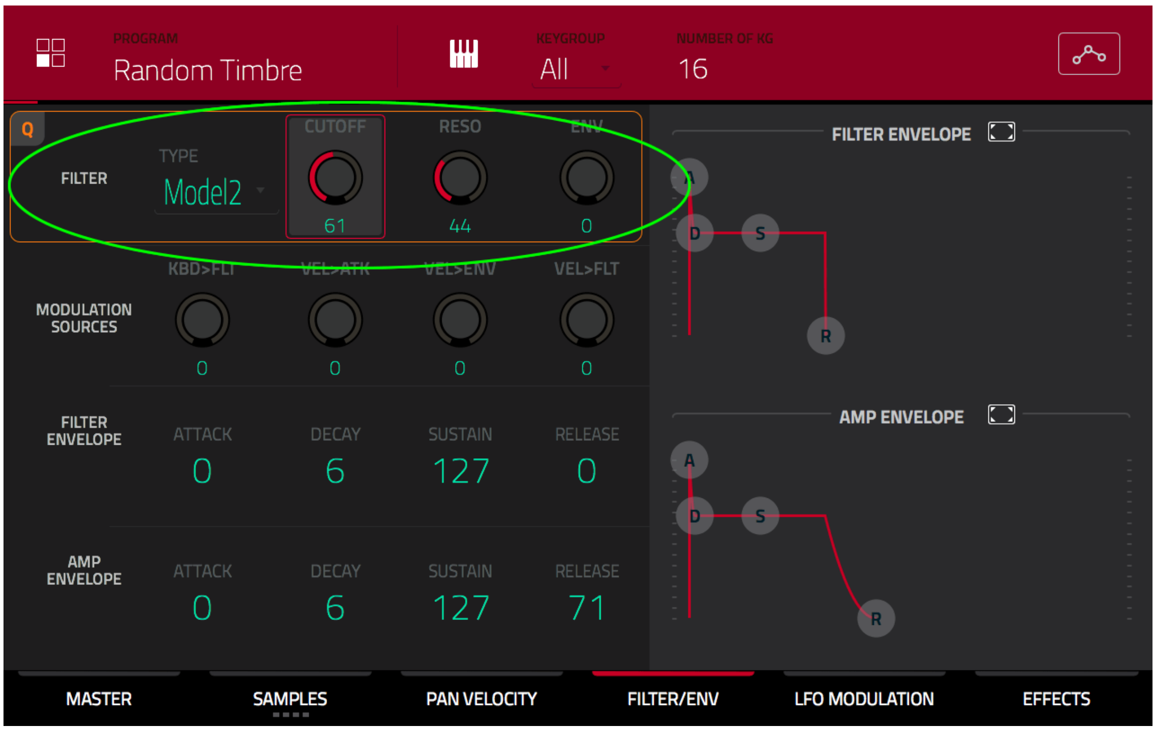
As you can see, both layers are set to cover the entire velocity range of **0-127**; **layer play** for all the key groups in this program are set to the default **VEL** (in the **MASTER** screen).

The key 'stereo' setting is found under the **PAN** column, where each layer is hard panned to opposing sides.

Play the program to hear the stereo effect in action. The advantage of this method is that the left and right channels sound slightly different as they are generated from different samples; they are the same pitch, just slightly different timbres, and this is what gives this pseudo stereo effect it's unique depth.

Filters and Effects

An acoustic piano can be transformed into all manner of different sounds just by using some of the built in features of a keygroup program. First, make sure **KG Select** is set to **ALL** so you are editing all the keygroups simultaneously. Head over to **FILTER ENV** and on the top row, change the filter **'TYPE'** to read **Model2** which is an analog modelling filter.



Now set the **CUTOFF** to **61** and the **RESO** to **44.** Preview the new sound with your pads - sounds like the start of a nice Rhodes emulation to me.

Avoiding Clipping

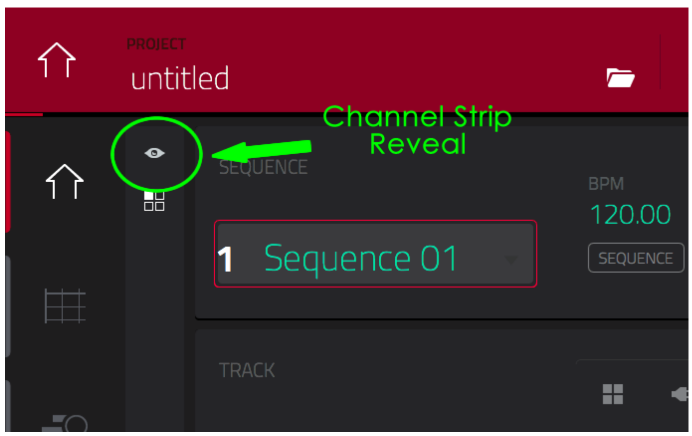
If you are hearing digital clipping when playing your notes, this is due to the low frequencies from each note sometimes overlapping and putting your global output level in the red. Fix this by reducing the output level of your program. In the **MASTER** screen, with '**Kg Select**' set to **ALL**, change the '**VOLUME**' for all keygroups to approximately **40**.

In my opinion a Rhodes sound needs some additional effects to make it come alive, so let's add some of the MPC Software's internal effects.

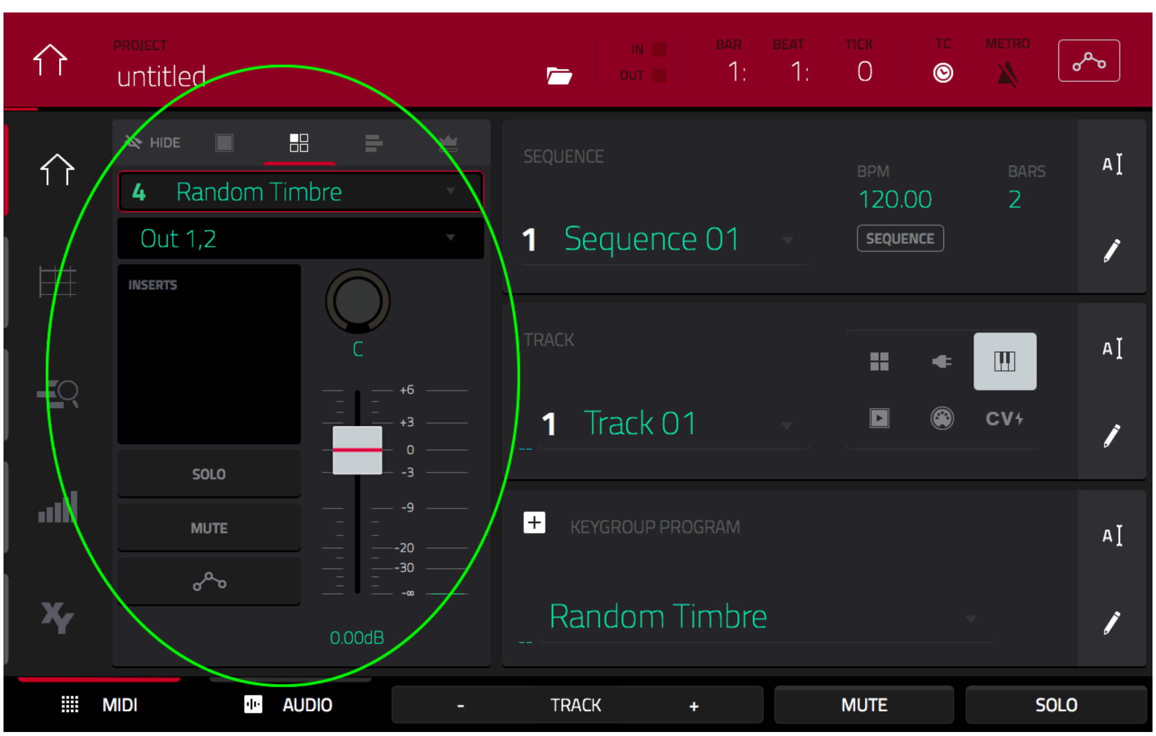
We've previously seen the **PROGRAM EDIT** > **EFFECTS** screen, which we used in a DRUM program to add 'insert effects' to a particular pad. In a keygroup program this screen works in a similar way, except it allows us to apply effects to a particular *keygroup*.

Now in this example, I want to add effects to *all* keygroups equally – while I could select 'ALL' keygroups to apply the same effect across all keygroups individually, a better way is to add the effects as a *program insert*. A program insert will apply insert effects globally to the entire program itself, thus affecting all keygroups.

Go to **MAIN** and press the small 'eye' icon in the top left of the screen:



This reveals the **PROGRAM CHANNEL STRIP**:



Tap inside the box labelled '**INSERTS**' to bring up the keygroup program inserts selection and double tap **insert 1** to bring up the effect selection window, and under '**Internal Effects**' choose the '**Tremolo**' effect. Tap the pencil icon to bring up the effect settings screen – configure the settings as follows:

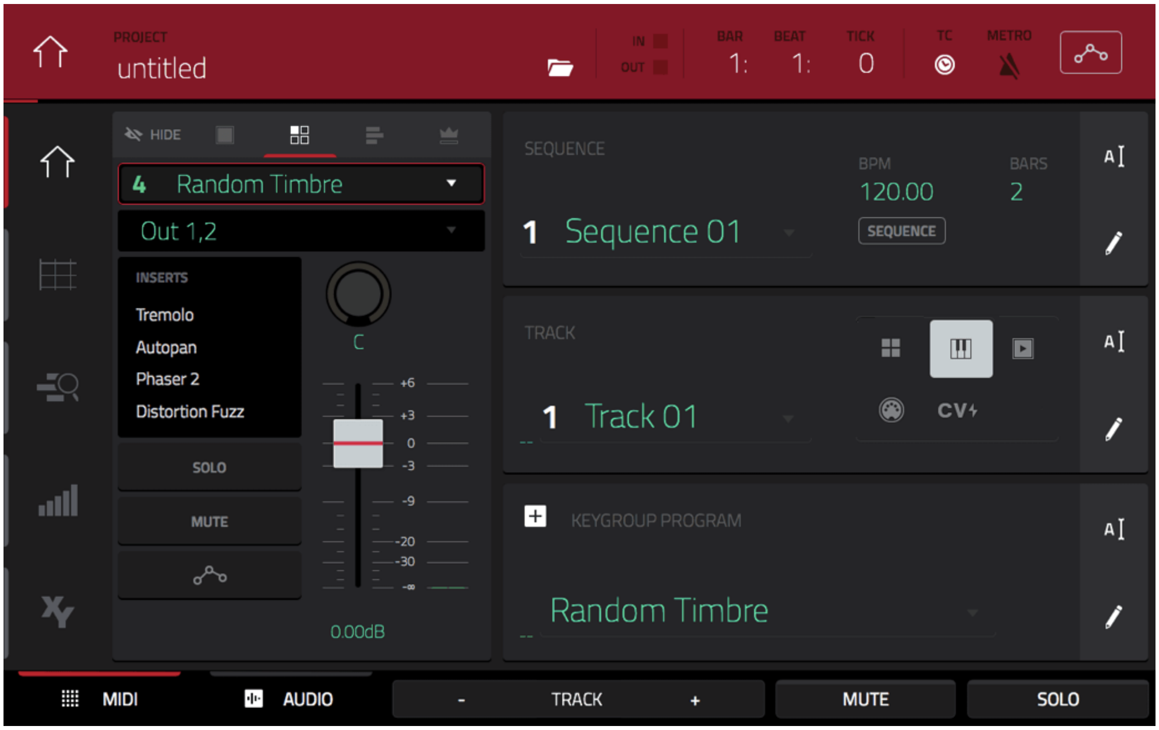
Try a **Rate** of **67** to give us a gentle, relaxing tremolo effect. You can leave the **Sin>Sqr** setting at **0** – basically increasing this gives a more square wave shaped tremolo action which is a bit harsh for an electric piano – at 0, you are getting a pure sine wave tremolo effect, which is more gentle.

The **Dry/Wet** option allows us to control the amount of original unaffected signal we allow to remain. If you feel the tremolo effect is a little over the top, drop the **Dry/Wet** value down to something more subtle**, such as 34** and then close the settings panel.

Tap **CLOSE** and now tap **Insert 2** and select the **Autopan** internal effect. Set the **Rate** to be **77**, and set **Dry/Wet** to taste. Listen to this on the headphones to hear the autopan in full motion.

For **Insert 3**, select the internal **Phaser 2** effect. The default options work well for me – the phaser adds a gentle phasing movement to your notes, it gives them a little more body and richness. You could also try using a **Chorus** here.

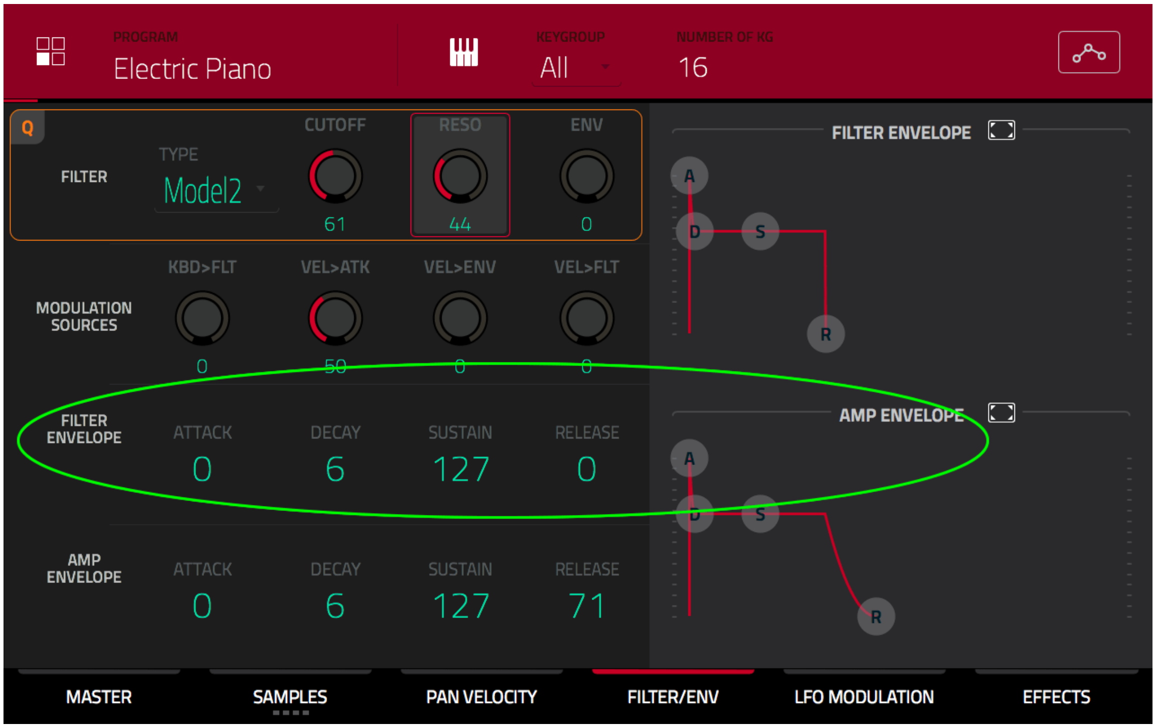
Finally for **insert** 4, select **Distortion Fuzz**, with a **Dry/Wet** of **21** and **Drive** of **6**.



Load up my version if you wish – **Electric Piano.xpm**

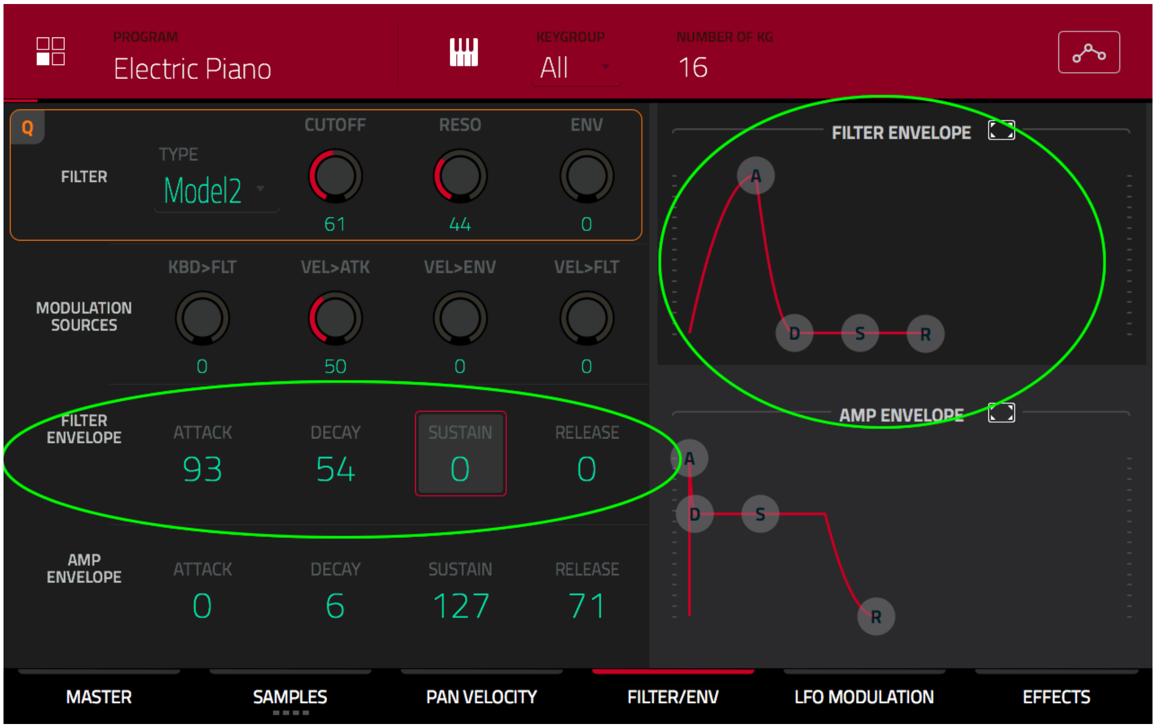
Utilising the Filter Envelope

We can start turning our program into something quite different by adding further changes, so it ultimately doesn't sound like a 'real' instrument at all. At this point, let's discover the *Filter Envelope*. Go to **PROGRAM EDIT > FILTER ENV**:



The filter envelope works in a similar way to the amp envelope in that it 'shapes' the filter being applied to each note you play, using a combination of ADSR (attack, decay, sustain and release). You also get a nice graphical display of the envelope in your computer.

To understand how the filter envelope works, let's try some extreme examples. So, first select **ALL** keygroups using the 'KEYGROUP' parameter in the top red bar of the screen. Now on the FILTER row, let's reduce the **CUTOFF** for the FILTER, just so the effect will be more obvious – set the **CUTOFF** to **61**. Now set up the following envelope settings for the **FILTER ENVELOPE** (third row); **ATTACK: 93, DECAY: 54, SUSTAIN: 0, RELEASE 0**:



Now, if you start playing your pads, you'll hear no change. This is because we have not yet told the MPC to 'take notice' of the filter envelope, so currently it is still just applying the Model 2 filter at a consistent rate across each entire note.

To allow the filter envelope to act on our sound, we need to increase the **ENV** setting on the FILTER row – this controls the amount of fully filtered signal sent to the filter envelope. In the first instance, turn it to the maximum setting of **127**, so that all the filtered signal goes to the envelope. Now pick a pad, say pad **D01**, and press and hold it. As it plays you will hear the filter envelope at work.

You should hear that the sound initially starts with a maximum filtered signal, i.e. as it was before we applied the envelope, but it then swells to a much more open filter sound. This portion is controlled by the **attack** setting of the filter, which as you can see is a fairly gradual increase.

The sound then retains that same open filter for a little while during the **decay** section which then forces it to plummet down to being highly filtered again (the **sustain** level of **0**). The **release** value is inconsequential when dealing with a zero sustain level as there's nothing left to 'release'.

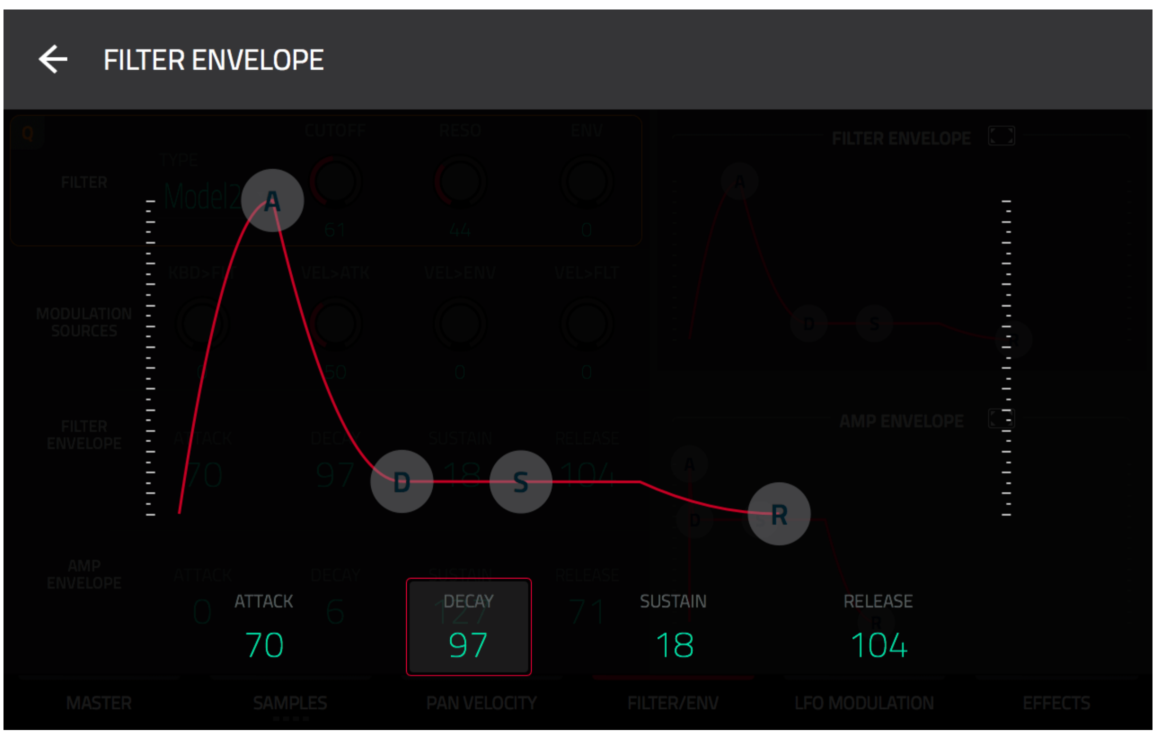
The key point here is that while the envelope waveform is at the bottom of the graph, the maximum level of filtering is applied (as if the envelope wasn't there). As the waveform heads higher in the graph, the filter is applied less until it reaches the very top peaks (127 values) where no filter is applied at all.

You can increase the attack to give a more gradual opening of the filter, and as you do this it will take longer to reach that plummet down to the 0 sustain level (in fact once you set an attack of 127, it seems to never reach any further).

Now obviously I set a rather dramatic drop to a zero sustain, so to create something more subtle we need to increase the decay setting as this will produce a more gentle drop to the zero sustain value. Try a **Filter DECAY** of **97**, but let's lower the **filter ATTACK** to **70** so that the envelope fade in acts much more quickly.

If you want to reduce the amount of filtering applied at the sustain level, you'll need to increase the sustain value – for example, try a **filter SUSTAIN** of **50** and hold that pad down so you can hear it working. I preferred a setting of **18** to keep the sustain quite well filtered.

Finally what about the **RELEASE**? This controls how the filter envelope will act the moment you release the pad. It's currently at 0, so when you release the pad the filter envelope will drop down to 0 instantaneously. Try jabbing at one of the pads a few times, only holding it down for a second – on the release, you should hear an abrupt 'squelch'. If you want to avoid that, set a more subtle release – I find a **filter RELEASE** around **104** should do the trick.



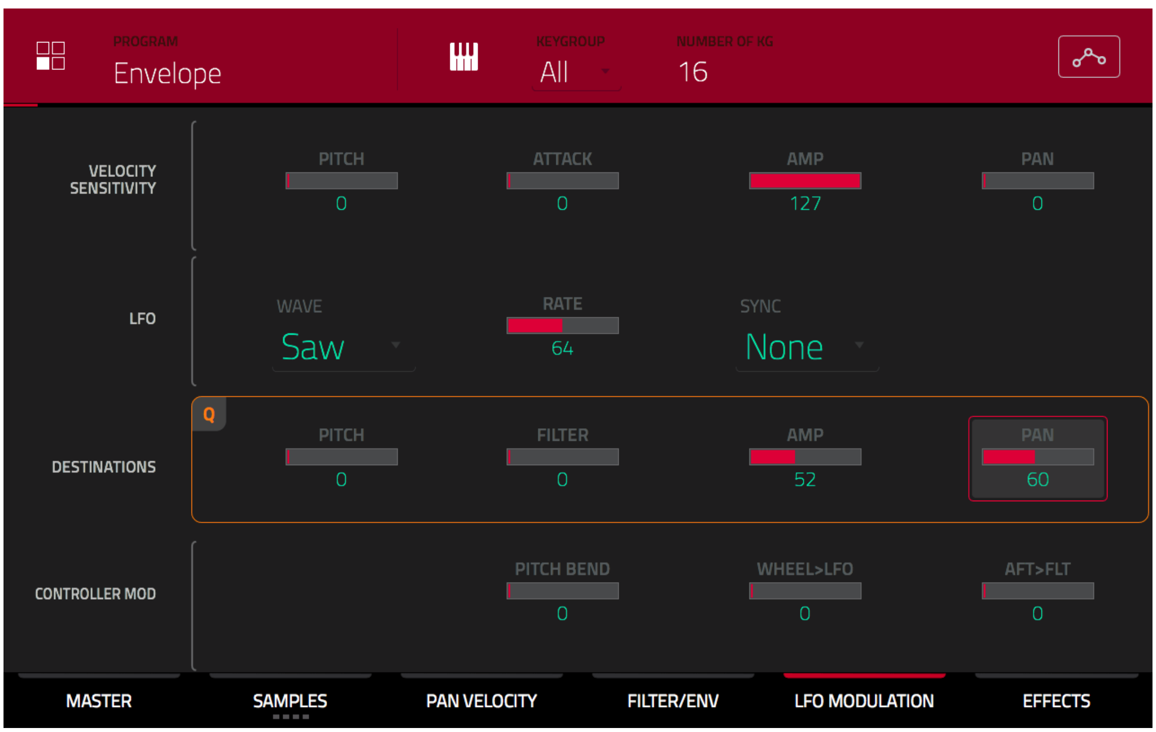
Finally, if you wish, consider dropping the '**ENV**' value from **127** to something a little more subtle, it really depends how obvious you want your filter envelope to act on your sound.

Load up the program **Envelope.xpm** for my version. Here I have further reduced the **CUTOFF** for the **Model 2** filter to **25** to really filter down the sounds.

Applying LFO

You can apply **LFO** to your keygroup programs in the same way as you would in a DRUM program. You'll have to be careful when it comes to the **L > PITCH** setting as you are dealing with musical notes, so changes in pitch could cause problems, although subtle use could be used to mimic vibrato for string instruments such as guitars. But the **PAN** and **AMP** settings can be used to great effect for all instruments.

For example (remember to have **ALL** keygroups selected), set **WAVE** type to **Saw**, and under **DESTINATIONS**, set **AMP** to **52** and **PAN** to **60**.



Try playing a few pads simultaneously to hear the new sound we've created. To make it even more complex, add a **FILTER** of **50**; now the filter is modulated by the LFO. In fact, go back to the **FILTER ENV** screen and increase the filter envelope **ATTACK** value to **127** and listen to the result. Was that ever an acoustic piano?

Load up **Experimental.xpm** to hear my version. I've also gone back to the *Amplitude* Envelope in the **FILTER ENVELOPE** screen and made some adjustments, increasing the **RELEASE** to **97** so, in conjunction with the LFO settings, we actually end up with a nice delay effect after releasing the pad. I've also increased **amplitude envelope ATTACK** to **64** to produce a more pronounced fade in of the entire sound itself.

You can obviously sit there and experiment with all the effects, LFO and filter settings all day to create endless new sounds. It may not sound like a piano any more, but it's definitely given you some new sonic textures to work with – so remember, no matter what type of instrument your keygroup program is initially set up for, there's always many opportunities to create many more unique instrument programs from it!

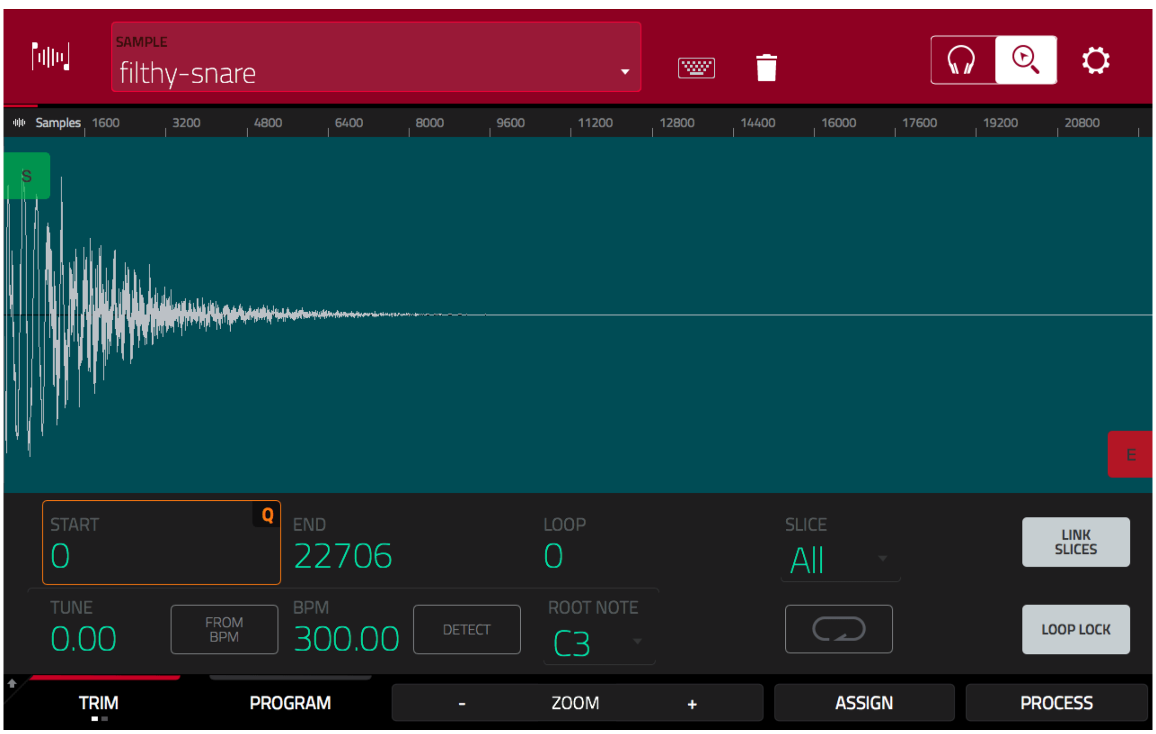
023 Build A Standalone Synth

Longing for a synth inside your standalone MPC? Not happy with the synth sounds available via Hybrid 3 in controller mode? Well don't forget that the MPC has plenty of filters, envelopes, LFO and effects, so it is more than capable of transforming itself into a pretty decent custom synth. We just need to feed it with the right type of sample!

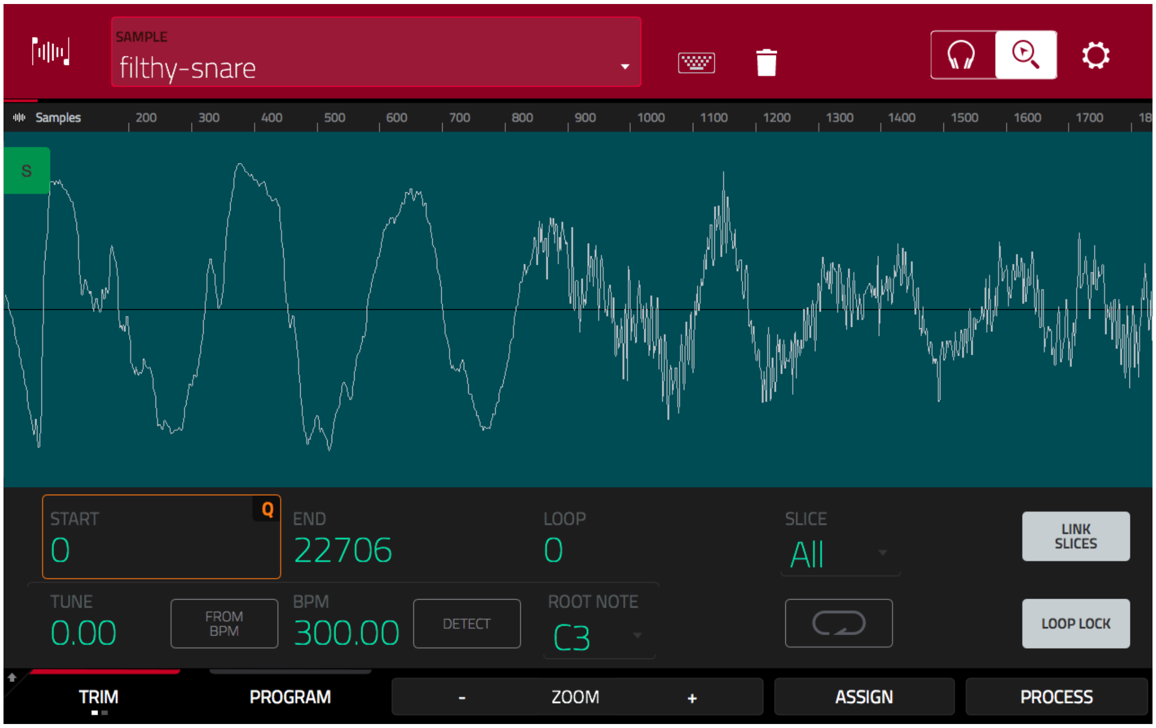
Creating The Looped Waveform

At the heart of our new synth instrument will be a short piece of audio. This can literally be from any source; for example, a portion of a drum sample, a tiny section from a vocal or acoustic instrument, or even a segment pulled from an entire song.

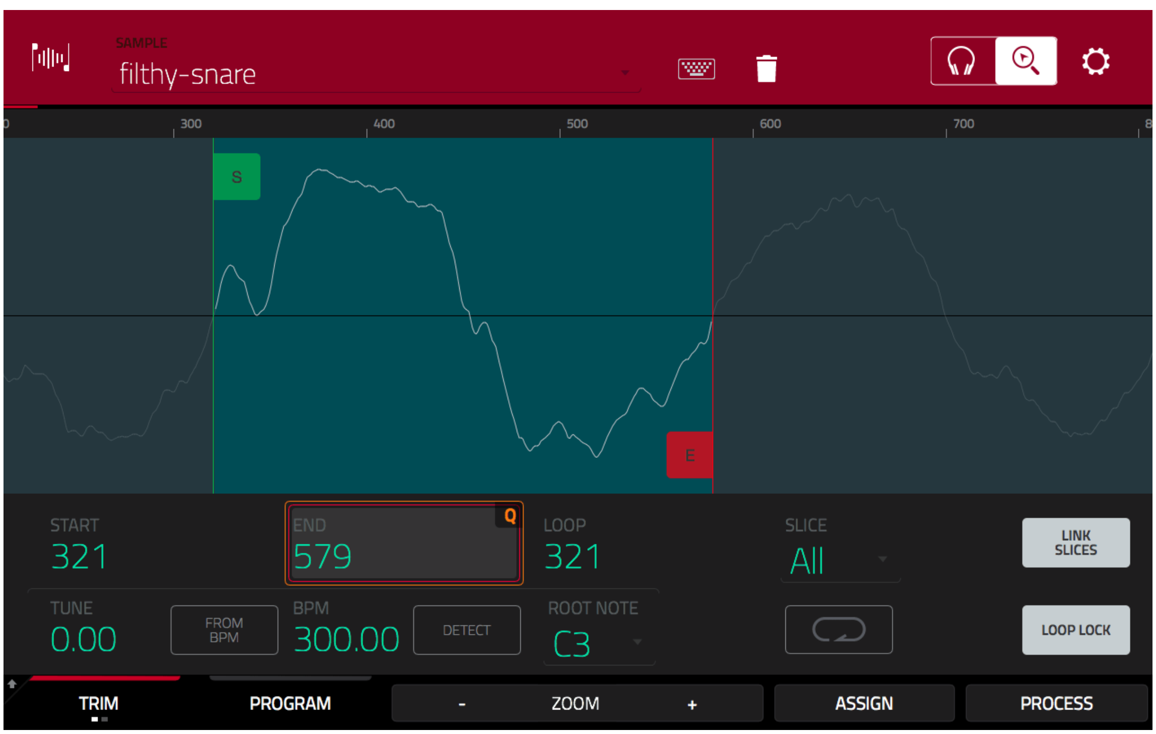
Load up the snare sample **filthy-snare** from the **chapter 023 folder** and take a look at it in **SAMPLE EDIT**:



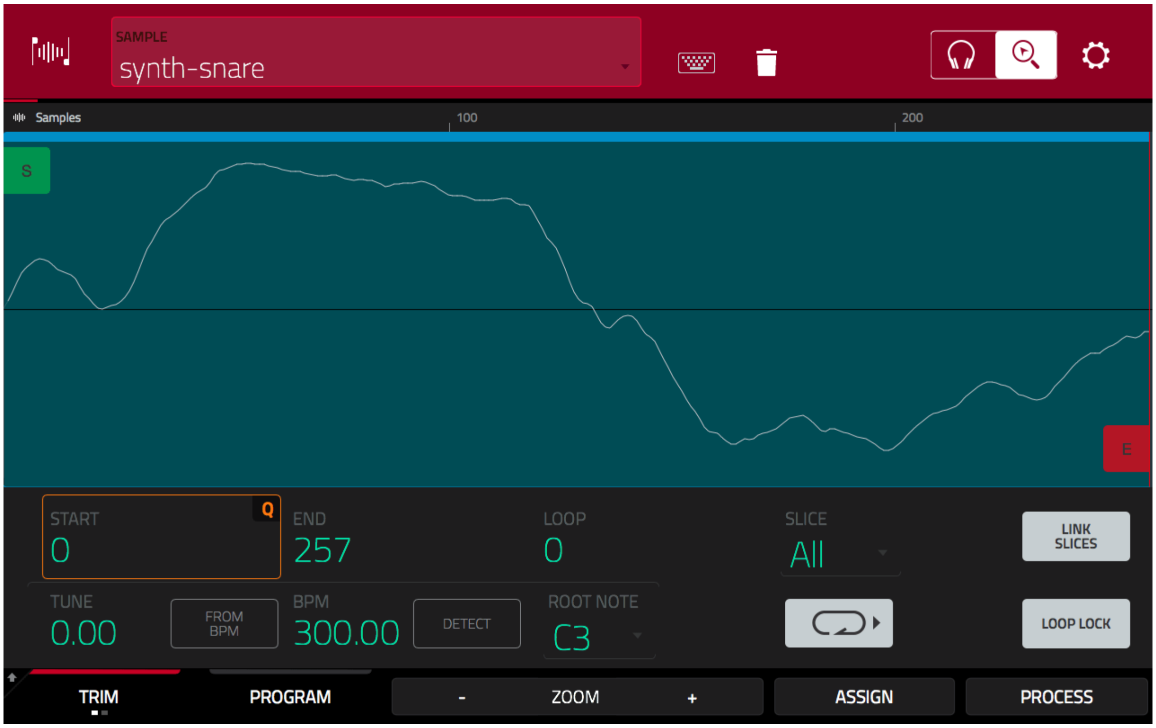
Let's take a very short segment from this sample and 'sustain loop' it using the looping techniques we've used previously in this book. Zoom in to the start of the waveform:



Let's loop that second waveform cycle. Hold down **SHIFT** and turn on **SNAP 0**; this ensures we get a click-free loop – try **START: 321** and **END: 579**. Preview your loop with **pad 16**.



Sounds good to me, so let's **PROCESS > DISCARD** the rest of the sample data, set **LOOP: FWD** and rename the sample **synth-snare**.



Tuning Your Sound

If you plan on using your synth patches with other instruments it's usually a good idea to tune your sample so it has a known musical pitch, and then set its root note accordingly. The MPC doesn't detect pitch automatically, so you'll either have to do this by ear or use a tuner.

To tune a sound by ear you'll need to load or play a reference sound while you adjust the TUNE parameter in SAMPLE EDIT until you get a match. This is quite a tricky thing to achieve in SAMPLE EDIT as this mode effectively only allows you to play one sample at a time.

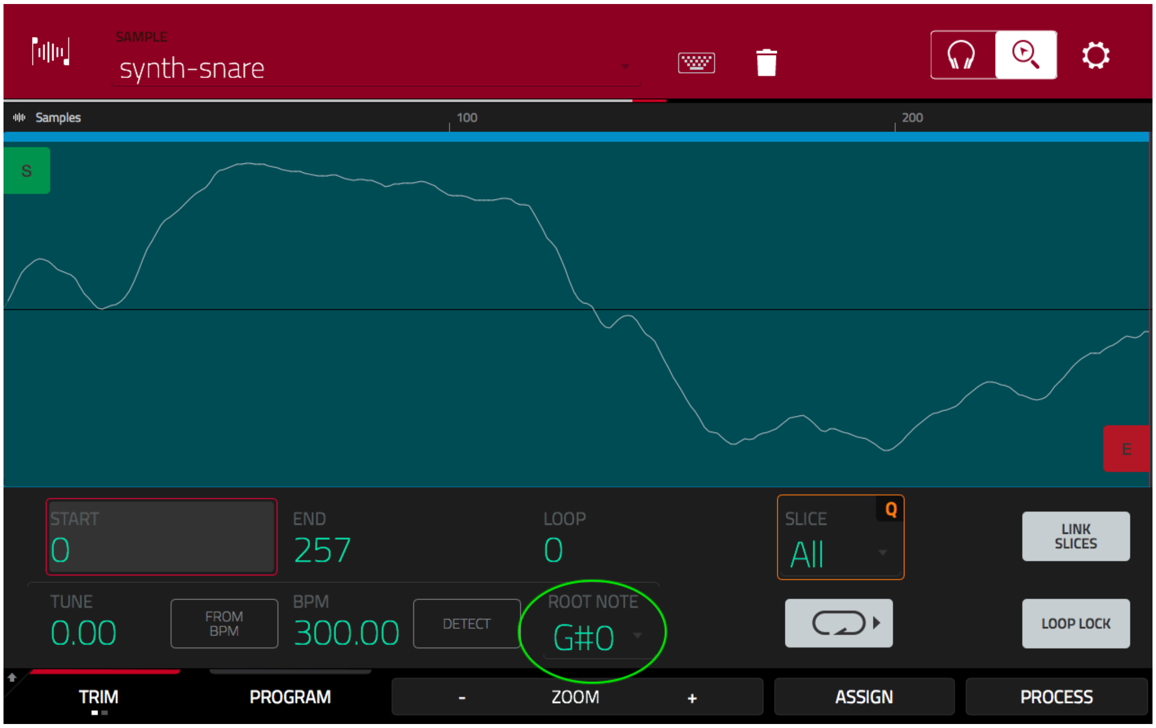
A workaround is to use a 'temporary' DRUM program. Load up the **piano-C1** sample into memory. This is a **C1** note; we'll use this as our reference note. In a blank DRUM program, assign this sample to pad **A01** and in **LFO MODULATION** set **SAMPLE PLAY** to **NOTE ON**.

Now assign the **synth-snare** sound to pad **A02** and again set **LFO MODULATION** to **NOTE ON**.

Now in **SAMPLES screen 2**, tap on the **SEMI** parameter and start adjusting this **SEMI** value for pad **A02**, increasing or decreasing until it matches the C1 note on A01. Keep alternating between A01 and A02 so you can hear the difference.

I eventually found a **SEMI** value of **+4** was required to make the synth-snare sample a C1 note (it might need some slight tweaking of the FINE parameter, but it's good enough for the moment).

So, if our synth-snare sample needs to be tuned up by +4, that means it's 4 semitones *lower* than C1, which makes it an **G#0** (G sharp 0). Head over to **SAMPLE EDIT** and set the **ROOT NOTE** to **G#0**.



Which Octave?

It's pretty hard to work out which octave our sample is as it's a very different sound to the piano reference file, but the main goal here is to find it's musical pitch. We can change the octave later (e.g to G#1 or G#2) by editing the root note or by adjusting the MASTER tuning of our entire synth program.

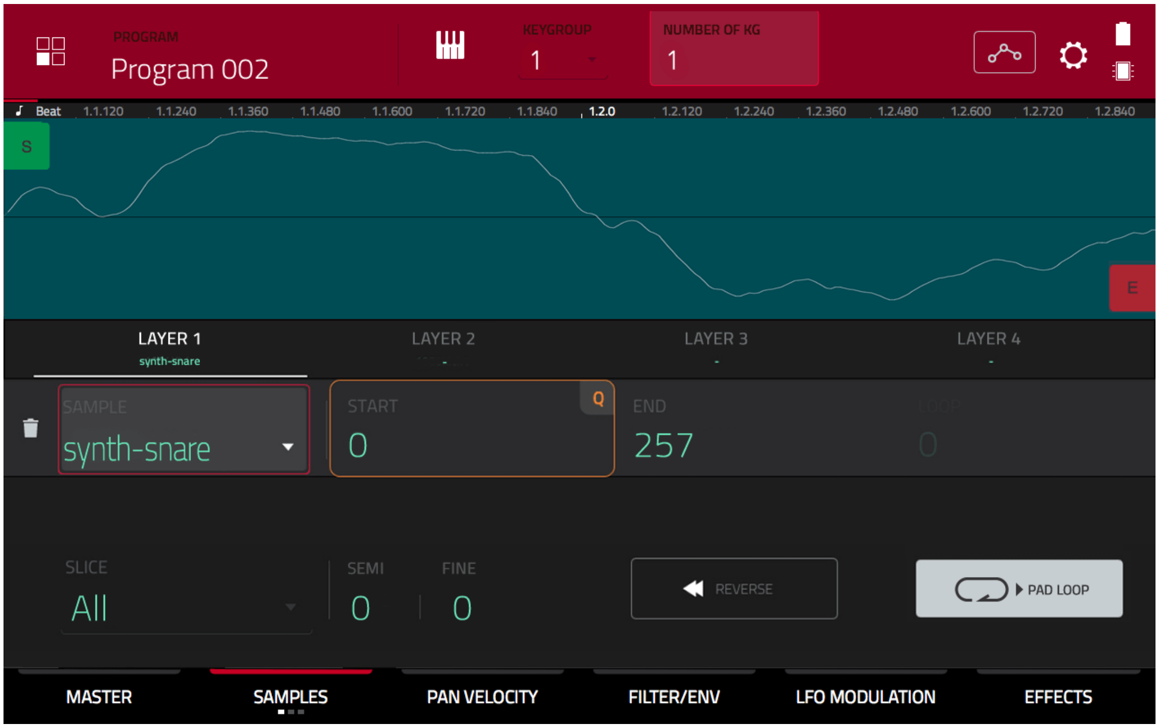
Now, our sample here is actually pretty close to G#0, but some sounds might fall somewhere in-between two semitones (G#0.5!) – in this case you'll need to 'fine tune' the sample's TUNE parameter in SAMPLE EDIT until it matches 'concert pitch'. I advise doing this once the synth program has been fully set up as you can play your synth program against a different keygroup program or plugin instrument and adjust the TUNE parameter while the sequence is playing.

Using Apps

Alternatively, there's hundreds of 'guitar tuner' apps for Android and iOS that will tell you the pitch of your sample, so you could play your sound into your phone's mic and get the app to tell you the musical pitch of your sample.

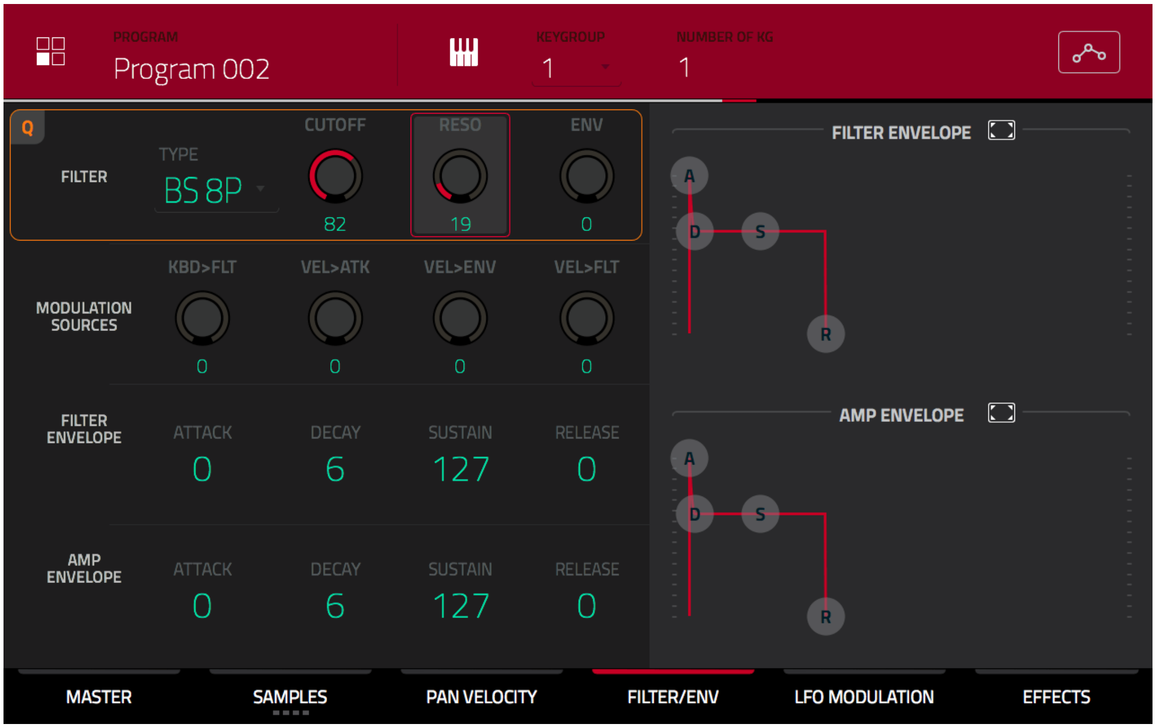
Building Your Keygroup Program

Go to **MAIN** and create a new **KEYGROUP** program. Go to **PROGRAM EDIT > SAMPLES** and assign the **synth-snare** sample to **layer 1** of the default keygroup.

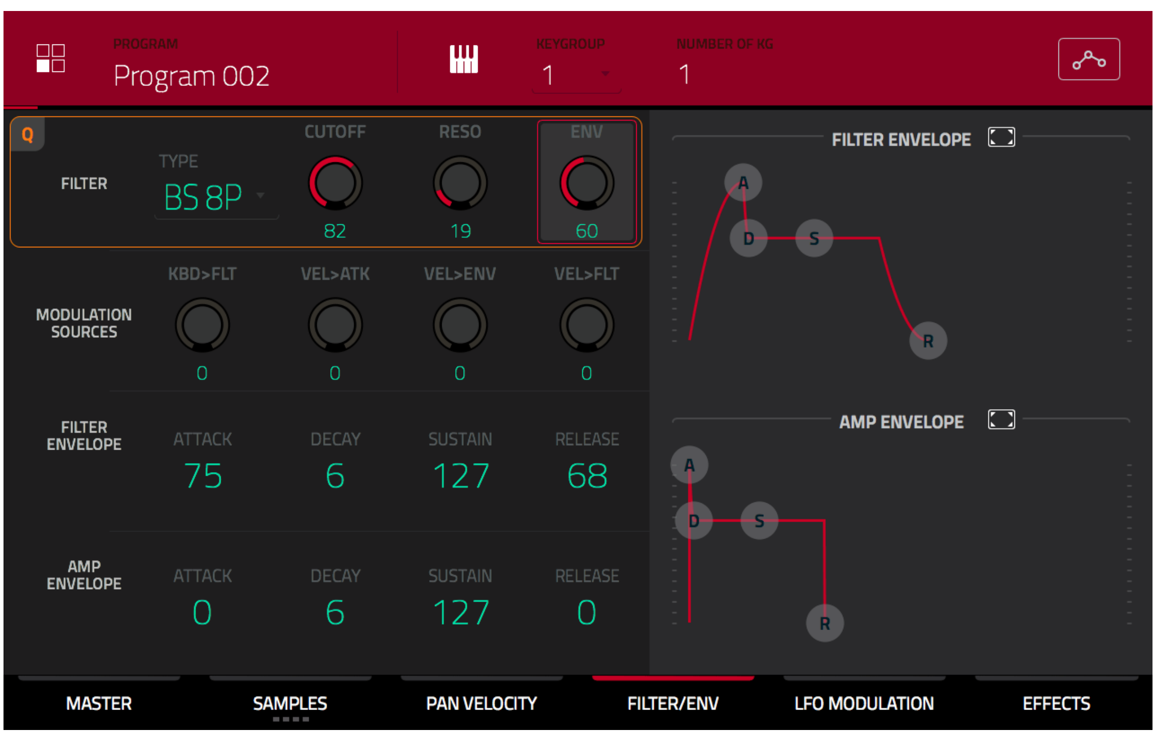


This will automatically spread our synth sample chromatically across all 128 MPC pads (C-2 to G8). Select **BANK F** and give it a try – play a few pads to hear the sample in action.

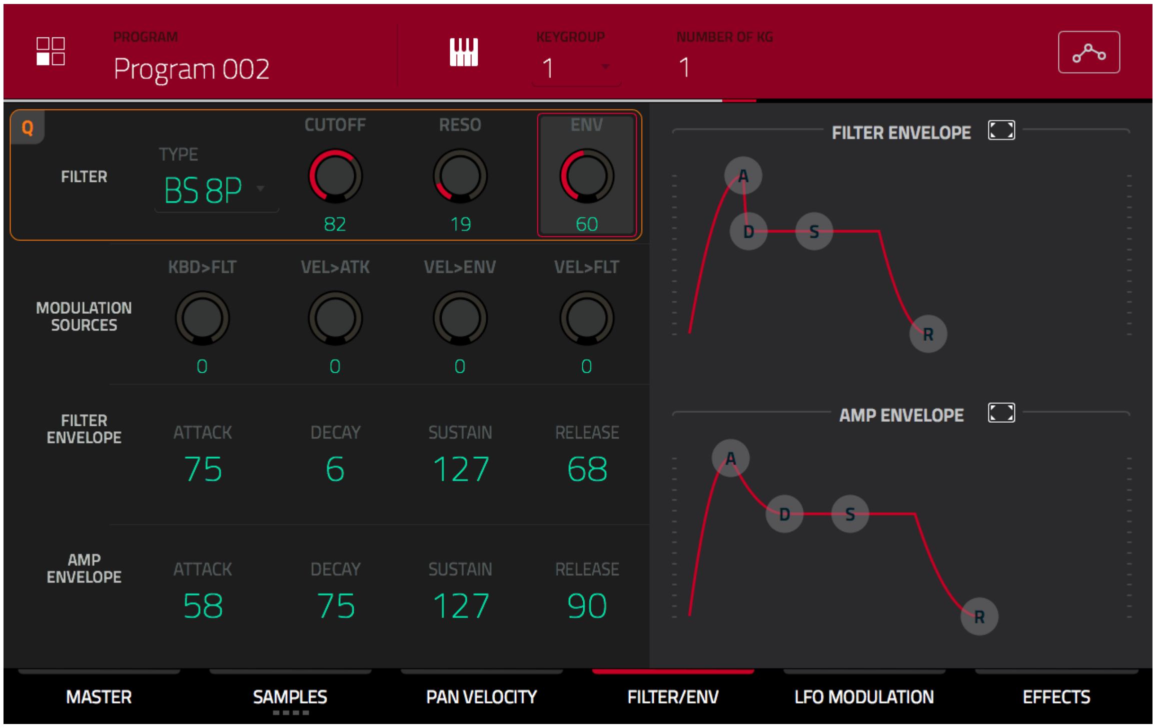
Now, let's add a nice filter to this sound. Try a bandstop, **BS 8P** with **CUTOFF**: **82** and **RESO**: **19**.



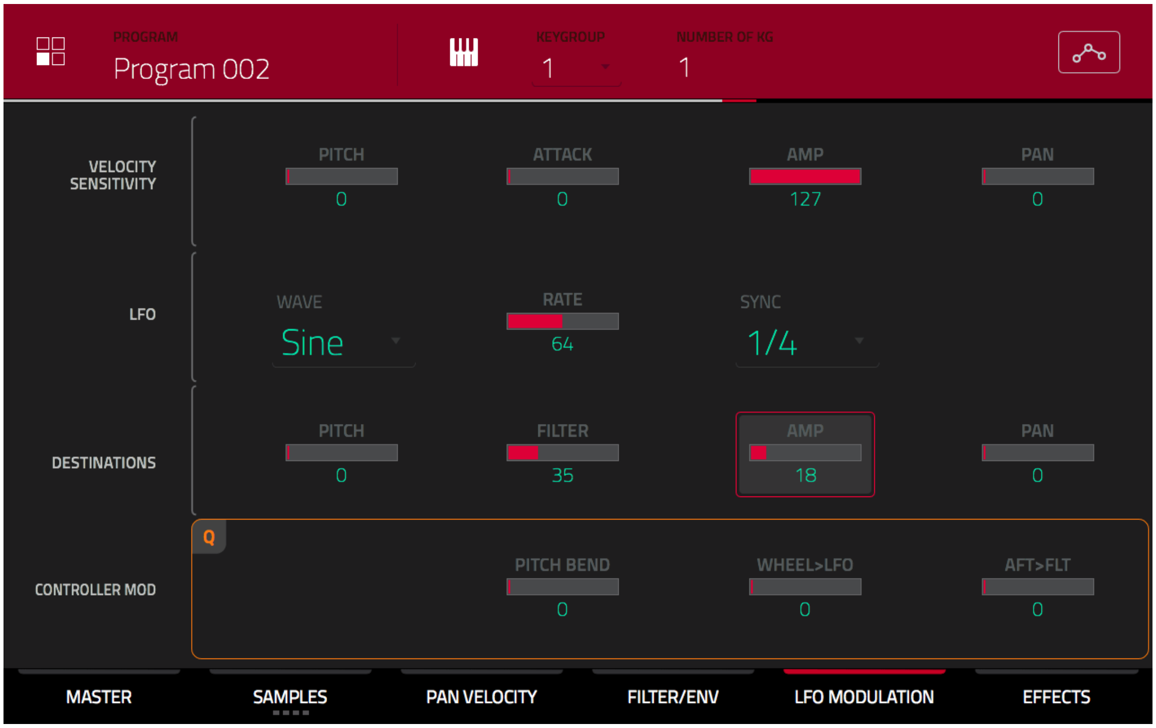
Let's shape that filter a little by adjusting the **FILTER ENVELOPE**. Set **ATTACK: 75** so the filter gradually comes in, and a release of **68** so when you release the pad the filter slowly fades out. Finally set the **FILTER ENV** to **60** to push some of the original signal via the filter envelope:



Currently the **AMP ENVELOPE** has zero release, so we're getting a horrible click each time we release the pad. For our synth sound I'd prefer a nice extended release, so set the **RELEASE** to **90**. While you're at it, increase the **DECAY** and just smooth the **ATTACK** by a small touch:



Let's breath some life into our bass sound by using the LFO. Go to **LFO MODULATION** and set the LFO to a **Sine WAVE** and **SYNC** of **1/4.** Under **DESTINATIONS** set **FILTER** to **35** and **AMP** to **18**:



That's a pretty awesome pad sound already! Let's not stop there though. Go to **PROGRAM EDIT > EFFECTS** and insert a **DELAY SYNC** from the internal effects. Configure the delay parameters to something like the following:

Now head back to '**PROGRAM EDIT > SAMPLES screen 2**' and assign a second instance of our **synth-snare** sample on **layer 2** of our keygroup. Change the **TUNE** parameter for **layer 2** only to **-5** and drop the **LEVEL** for both layers to compensate for the increase in volume:

The tuned down layer has effectively created a simple, *harmonised* chord for each note in our program.

Finally head over to **PAN VELOCITY** and pan each layer to opposite sides to achieve a nice stereo effect. I've set my **PAN** to **35L** and **35R** but you should experiment to achieve the stereo width you prefer.

Rename the program to **Ethereal Pad** – you can load up my version from the tutorials folder, **Ethereal Pad.XPM**.

From a humble snare sample we've created our own unique synth patch and you can of course continue to experiment with program parameters to create different patches based entirely on this sound alone. Load up the **Funky Lead.XPM** program to hear a monophonic lead patch (**MASTER > POLYPHONY > MONO**) that uses the same sample but with completely different program parameters.

The program **Growly Bass.XPM** is also made from the same snare loop – try this one in BANK B to hear a very dirty bass sound. Check out all the different program parameter screens to see exactly how I achieved this – notice I tuned this entire program down by one octave (**-12**) in **PROGRAM EDIT > MASTER > TUNE**.

And remember, you can of course use different looped waveforms as the core sound source of your instruments, so there literally is an infinite number of possibilities.

024 Building Songs & Performances

After creating your various programs, sounds and individual sequences, you'll no doubt be itching to put them all together as part of a complete composition or performance.

Working on the Basic Song Structure

Load up the project file '**RPG'**, select **sequence 1**, '**RPG – Core Template**' and press **PLAY START**. This sequence represents an initial idea for a complete song; it's currently only 8 bars long but it contains all the core elements I'll need to build a song around.

Currently the sequence is built using four programs; '**RPG Drums'** contains our drum kit, '**RPG Sounds'** contains some musical phrases and sound fx, **Acoustic Bass** is a double bass keygroup program and **Old Piano** is a multisampled piano keygroup program.

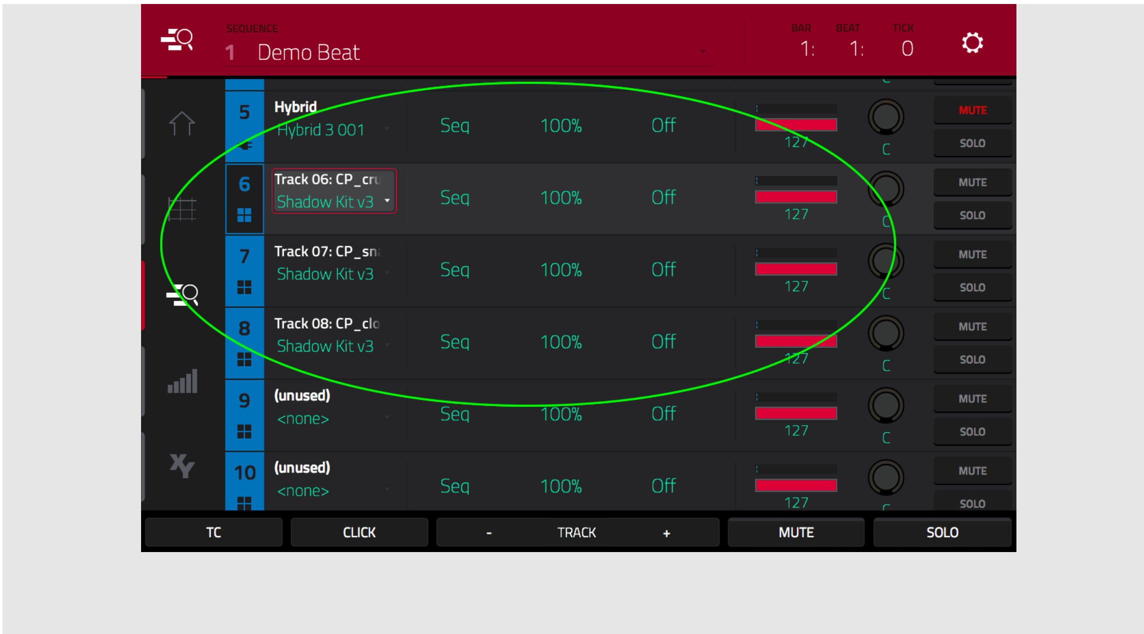
There's 8 tracks in total, with each dedicated to a particular sound, so everything is nicely separated and full controllable at both program level and track level. You can solo each track to hear what each instrument sounds like (quickly done either via TRACK VIEW).

Exploding Tracks

You may have noticed that unlike the 'Demo Beat' sequence I was using earlier in the book, in this sequence I've placed each drum instrument on its own track rather than having a single track for 'Drums'. I tend to prefer splitting the drums like this as I can have a bit more control over the individual mixing of each drum sound.

If you've recorded your drums to a single track it's very easy to split each drum sound into individual tracks later. In **MAIN**, hit the **pencil icon** at the end of the **TRACK row** and select **EXPLODE**.

At this point the original track will be muted and you'll see new tracks appear – each track will be named after the sample they represent and will only contain events relating to that particular pad.



I want to keep this song fairly simple, just so we can get to grips with basic song building, so we're going to make the following structure

INTRO – a 4 bar intro

VERSE 1 – 24 bars

CHORUS – 8 bars

VERSE 2 – 16 bars

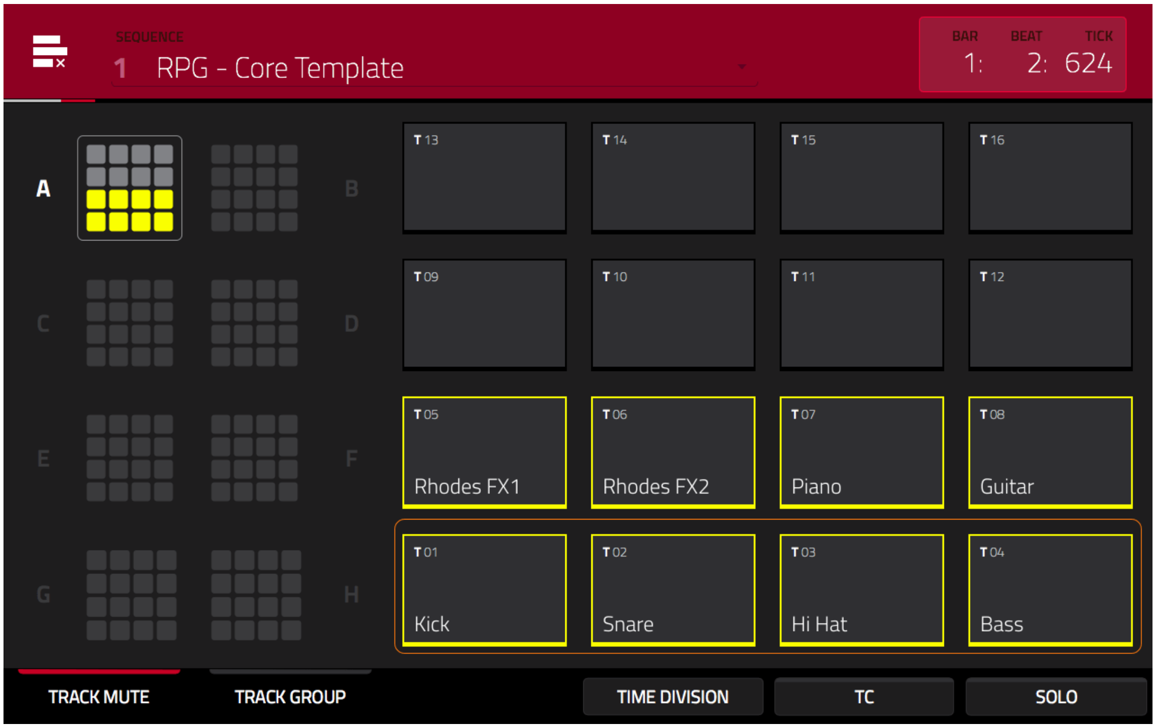
CHORUS – 8 bars

ENDING – a customised chorus that ends the song

I want each section to have a unique quality while maintaining the core musical concepts. To do this we can use a few different techniques such as recording additional events such as drum fills or variations in the bass line or melody line, completely re-recording certain tracks to change the performance entirely, or perhaps the simplest method and the one we'll look at now; selectively *muting* existing events.

Experimenting With Track Mutes

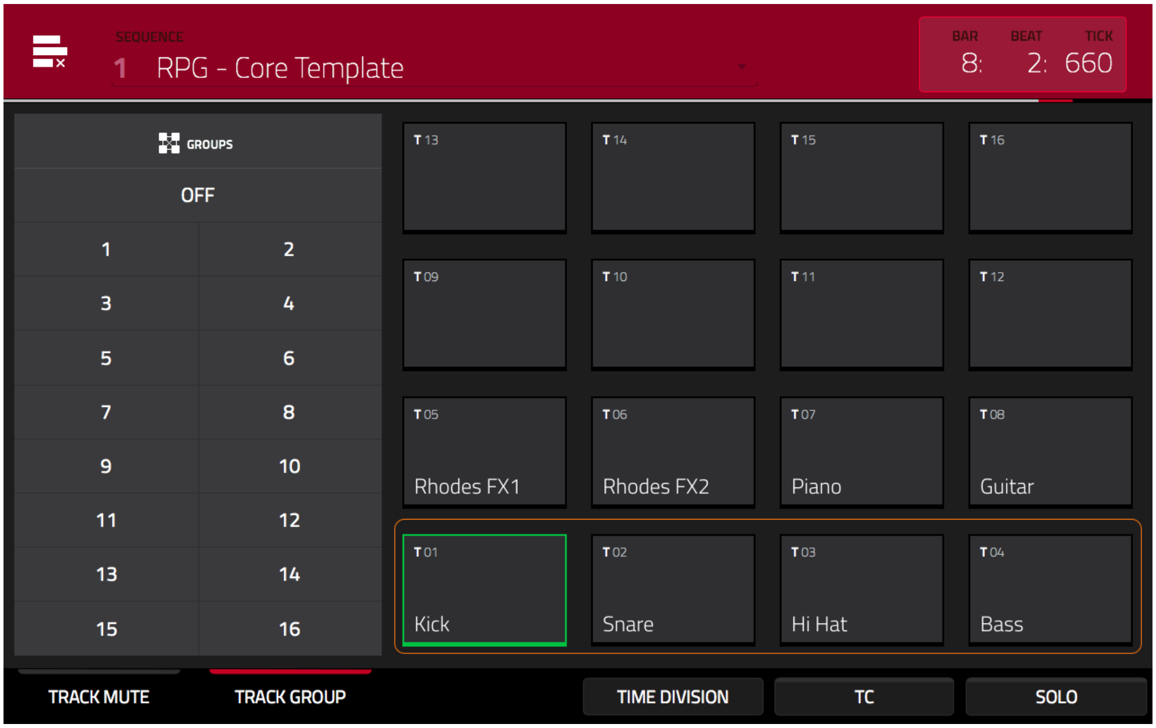
Go to **MENU > TRACK MUTE**:



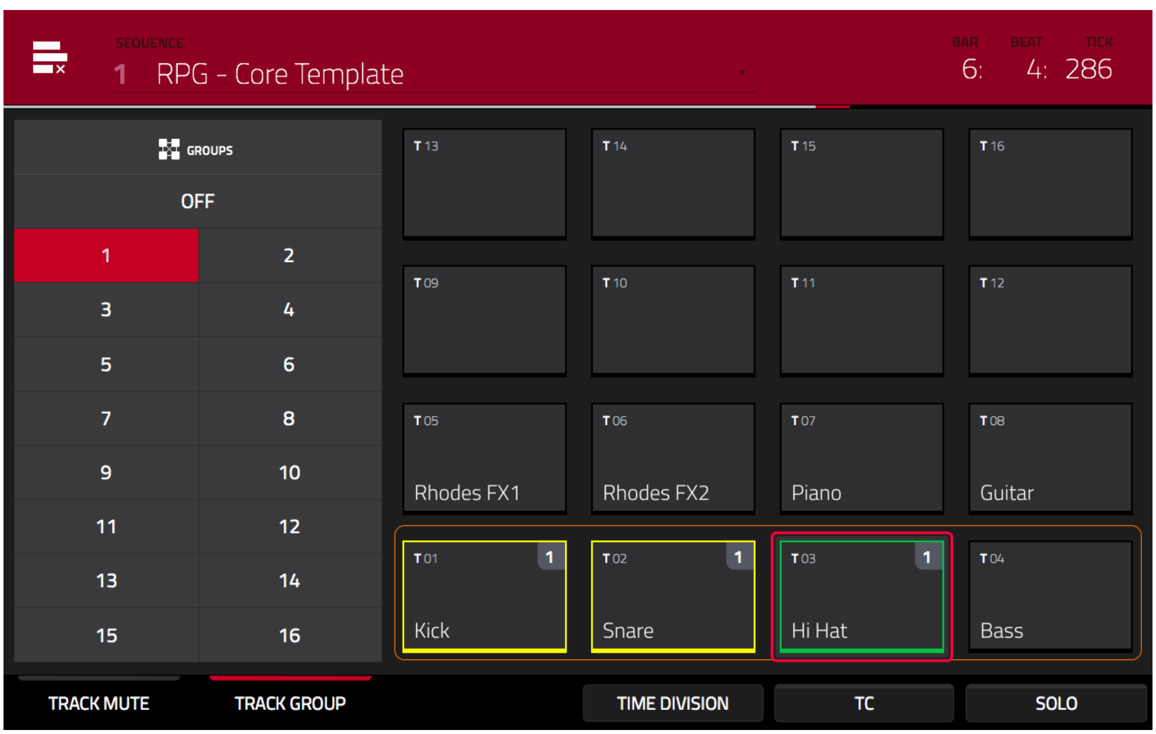
The TRACK MUTE screen allows you to easily mute and unmute any track in your sequence simply by hitting pads. Press **PLAY START**. Currently our 8 tracks are each assigned in order to one of the pads in BANK A – to mute a track, just hit the pad that corresponds to that track. Hit **pad A02**; this will mute the snare track. Hit **A02** again to unmute it. You can hit the actual pad or if you prefer you can hit the onscreen pad.

Now try muting then unmuting other tracks. As you can see, this is a very easy way to experiment with the content of a sequence.

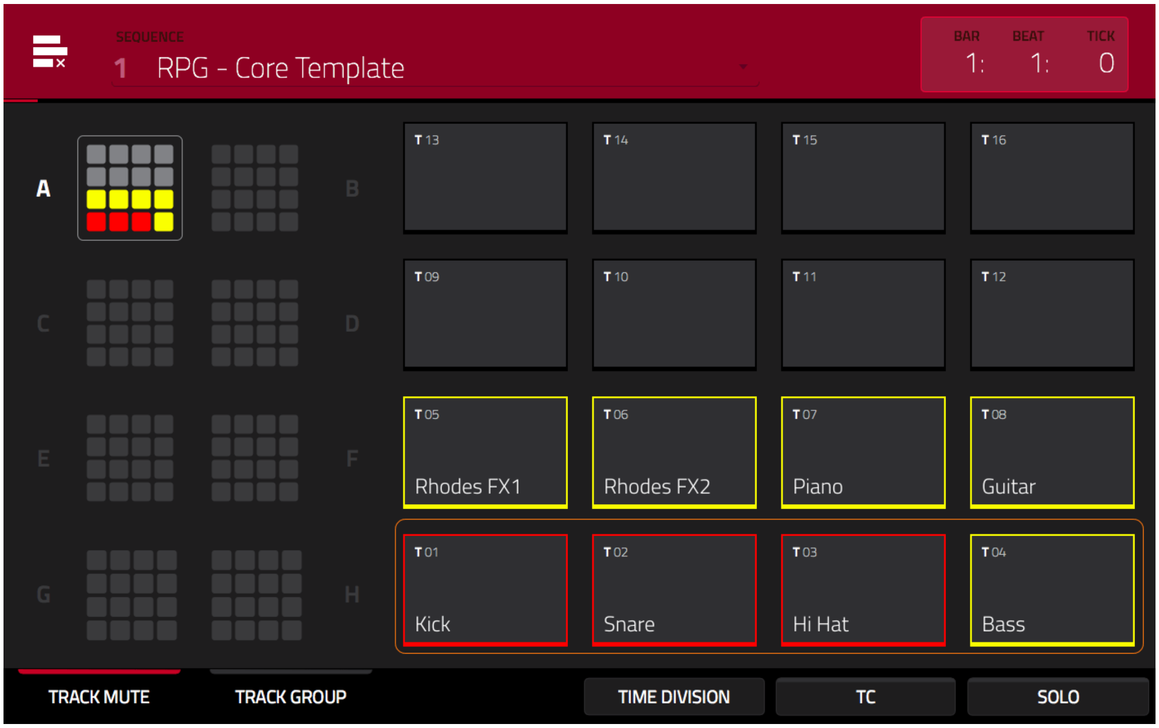
You're not limited to muting a single track at a time, you can mute as many as you wish, either gradually or simultaneously. To mute multiple tracks at the exact same time, the most reliable method is to set up a track group first. Press the **TRACK GROUP** button to enter the **TRACK GROUP** screen:



Let's put all the drums in the same track group. Tap pad **A01** and on the left side of the screen, tap the '**1**' button to assign the kick track to group 1. Do the same with the snare track and hat track, assigning these to group 1 also.



As you can see, the assigned track group for each track is now shown in the top right of the on screen pad. Return to the **TRACK MUTE** screen, press **PLAY START** and now press any one of the drum tracks (it doesn't matter which one) – upon doing so, all your drums are muted simultaneously.



Try spending a few minutes experimenting with track mutes to see what different muting ideas you can come up with. In the next section we'll use mutes to begin creating some of the primary elements of our song.

Creating The Chorus

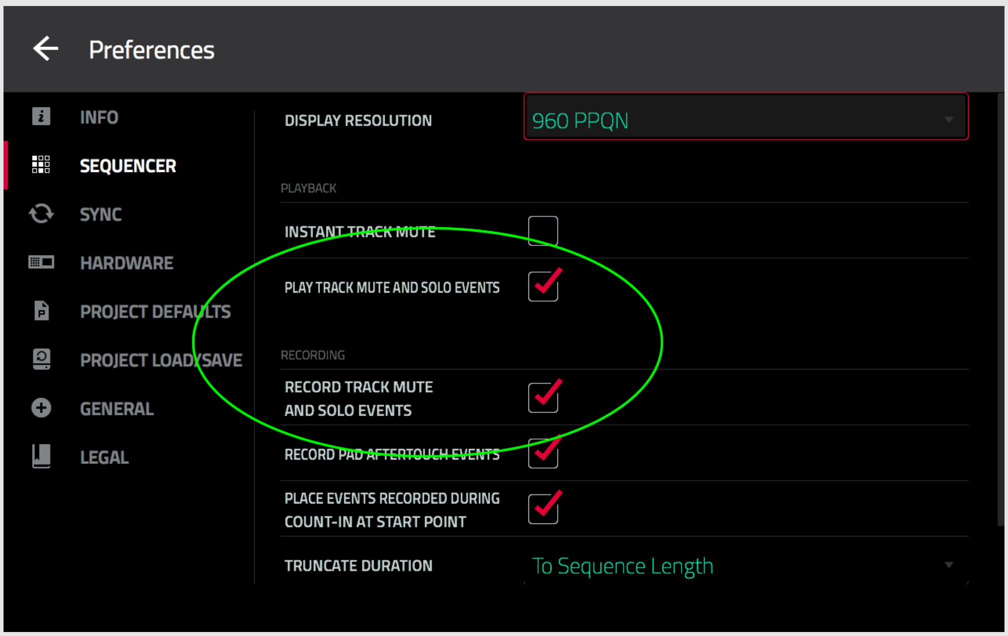
Let's use our core template sequence, **sequence 1 (RPG)** with a slight modification to help create our chorus – it's got a catchy little piano hook and a nice little guitar fill, so perfect for a chorus.

First, make a copy of **sequence 1** to the unused **sequence 2** and rename it **'RPG – Chorus'**. What I'd like to do is temporarily mute all the drums after the kick plays at 008.01.000, giving a nice 'pause' to the rhythm of the track – but most importantly I want to *record* these mutes to our sequence.

With our kick, snare and hat all assigned to track mute group 1 we'll be able to mute all three simultaneously just by hitting one of the associated pads. Press **OVERDUB** and **PLAY START** so our sequence begins playing back in 'overdub' mode. Remember, overdubbing leaves all existing events untouched, it's just going to let us add further events to our sequence; this time instead of note events, we'll be adding **track mute events**.

Recording Track Mute Events

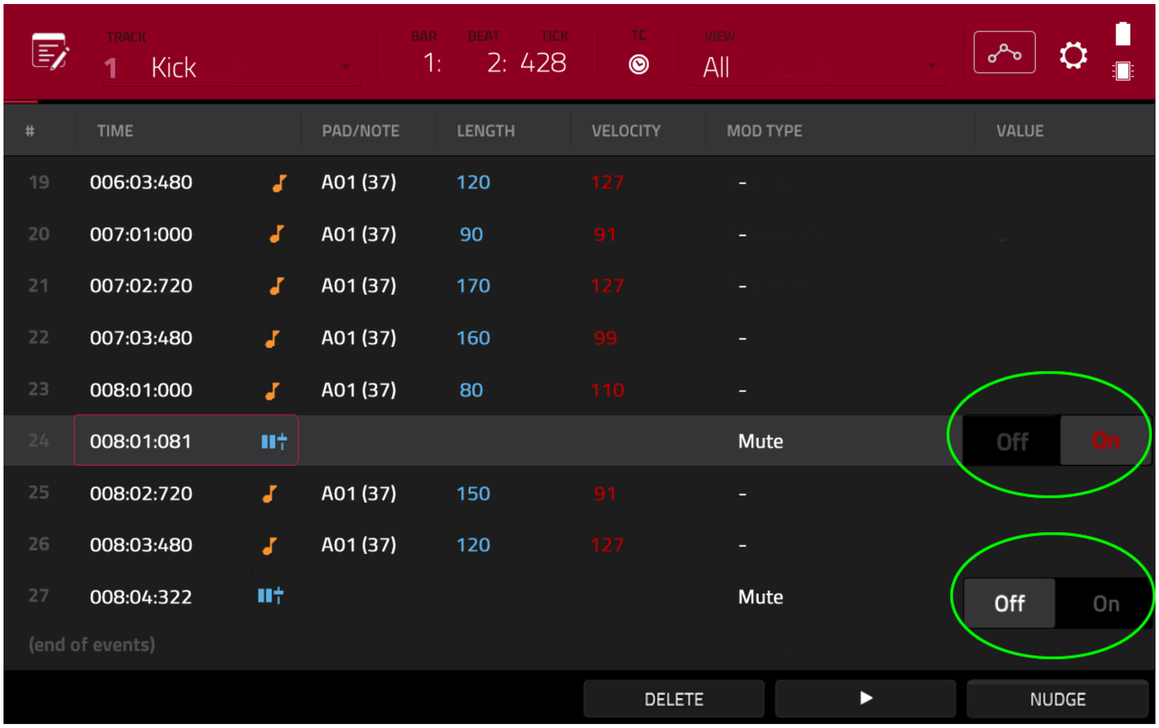
Before we try recording track mutes, go to **PREFERENCES > SEQUENCER** and make sure both **RECORD TRACK MUTE AND SOLO EVENTS** and **PLAY TRACK MUTE AND SOLO EVENTS** are enabled:



As soon as your sequence reached the start of the 8th bar, let the initial kick/hat play and then hit pad **A01** to mute all our drums. Then just before the sequence loops back to the beginning, hit pad **A01** again to unmute the drums. Press **STOP**. Now press **PLAY START** to hear your track mute in action.

Check out my version on **sequence 3** (**RPG – Chorus**).

You can edit the resulting track mute data via **LIST EDIT** – as you can see below, it's simply a case of switching the '**MUTE**' event **ON** or **OFF** on each track you wish to mute by tapping the on-screen 'mute switch'.



However it is currently not possible to *add* MUTE events via the LIST EDITOR. This can be done in the computer GUI, but in the hardware the only way to add track mutes is to record the mutes in real time.

*You can also delete all track mute events within a specified time range using the **ERASE** button; **ERASE > AUTOMATION,** however this will also delete all other automation data that you may have recorded within that time range, so this might not be ideal. I'll look at how we record automation data very soon when we use the XYFX feature.*

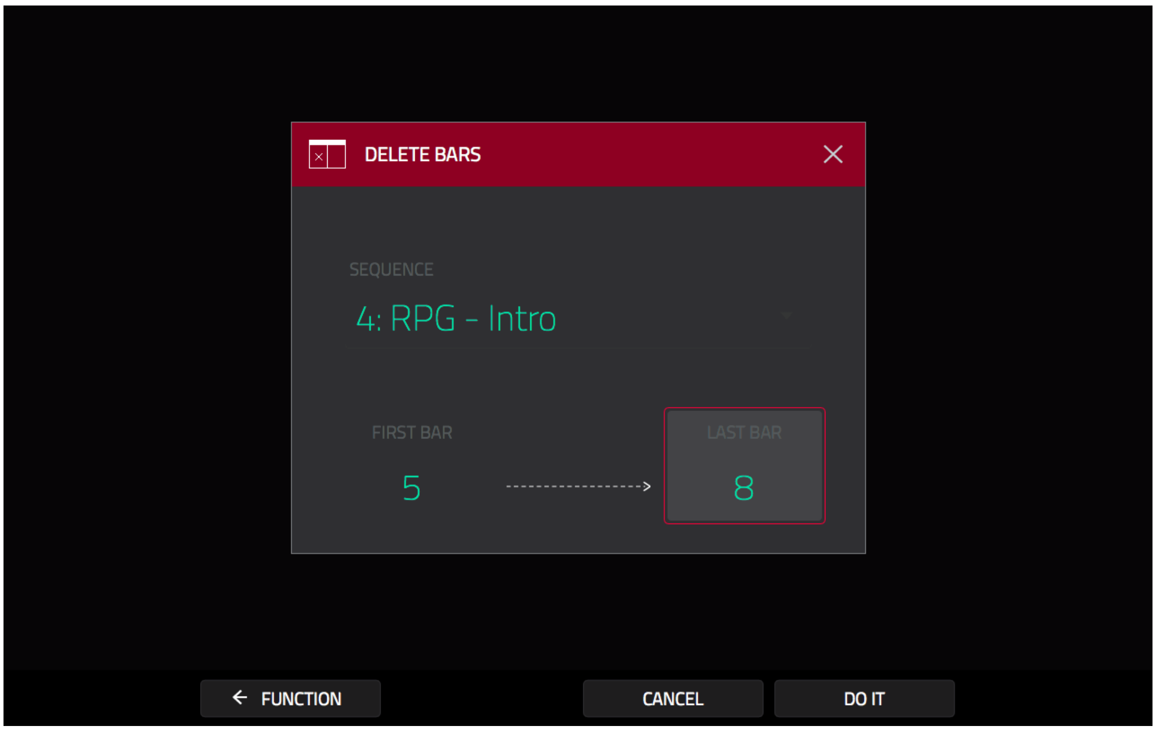
With our 'chorus' sequence now created, we can move on to creating the other sequence structure for the song.

Pad Mutes

An alternative to track mutes are pad mutes, which can be controlled via **MENU > PAD MUTE**. This screen works identically to the track mute screen, but this time you can control the muting of individual pads rather than tracks. Generally speaking you would use pad mutes in situations where your tracks contain mixed event data, for example where the kick, snare and hat had been recorded to a single track.

Creating The Intro

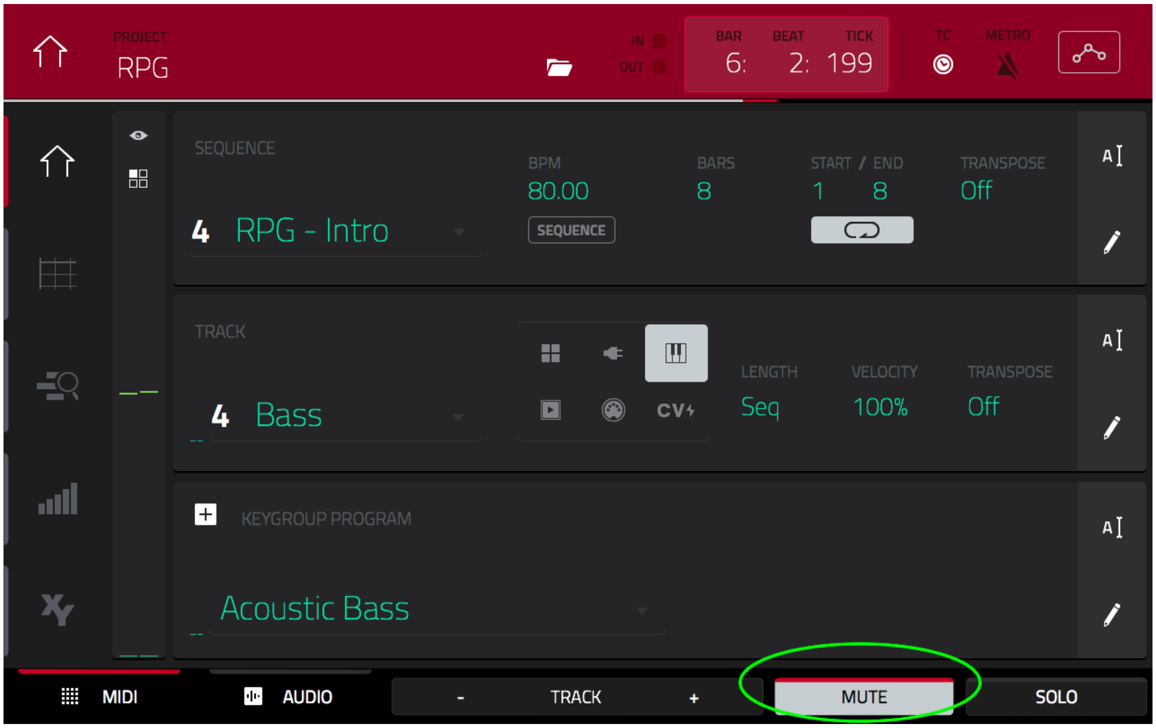
For the intro, I just want a 'stripped back' 4 bars, so **copy** our template **sequence 1** to blank **sequence 4** and give it a name '**RPG – Intro**'. In MAIN press the pencil icon on the **SEQUENCE** row and select **DELETE BARS**:



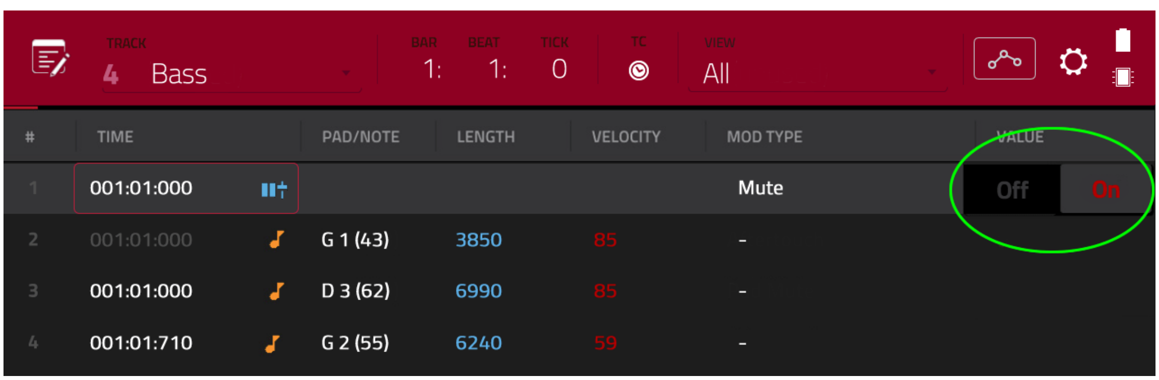
Set **FIRST BAR** to **5** and **LAST BAR** to **8** – this will delete bars 5-8 of this sequence. Hit **DO IT**.

For this intro I want to remove the bass line on track 4; you could press **ERASE** and just erase all events on this track, manually delete events in GRID, or we can keep the events and instead add a track mute to the entire track.

In **MAIN**, select **track 4** (**Bass**) and hit **OVERDUB** and **PLAY START**. During the count in period, press the **MUTE** button at the bottom of the screen.



Once the sequence begins playing, you can just hit **STOP**. As the MUTE button was active as the sequence started, the MPC will record a track mute event at the very beginning of this bass track. We can see this in the computer GUI list editor:



Turn off the **MUTE** button and press **PLAY START** and you'll see the MUTE button turns itself on and mutes the bass line.

Repeat the above process to completely **MUTE** the guitar (**track 8**).

Now I still want some piano here, but I just want the chords, no melodies. So select **track 7** (**Piano**) and hit **OVERDUB** and **PLAY START**. After the first chords play, hit **MUTE** so the subsequent melodies that follow it are muted. Just before you get to the start of the third bar, turn the **MUTE** off so the next chords play. And once again press **MUTE** to mute out the melodies.

Remember you can of course head over to **GRID**, select the **ERASE tool** and delete all the unnecessary notes one by one – the advantage of adding MUTES is that it preserves the original event data should you ever wish to recover it; it also has a ‘performance’ value, as you are effectively ‘playing’ the mutes, turning them on or off in a creative way.

So, that takes care of the piano melodies. Finally, let’s completely kill all the drums at the end of the bar, just after **4.01.000** – you could use the **TRACK GROUP** method that we used earlier in this tutorial.

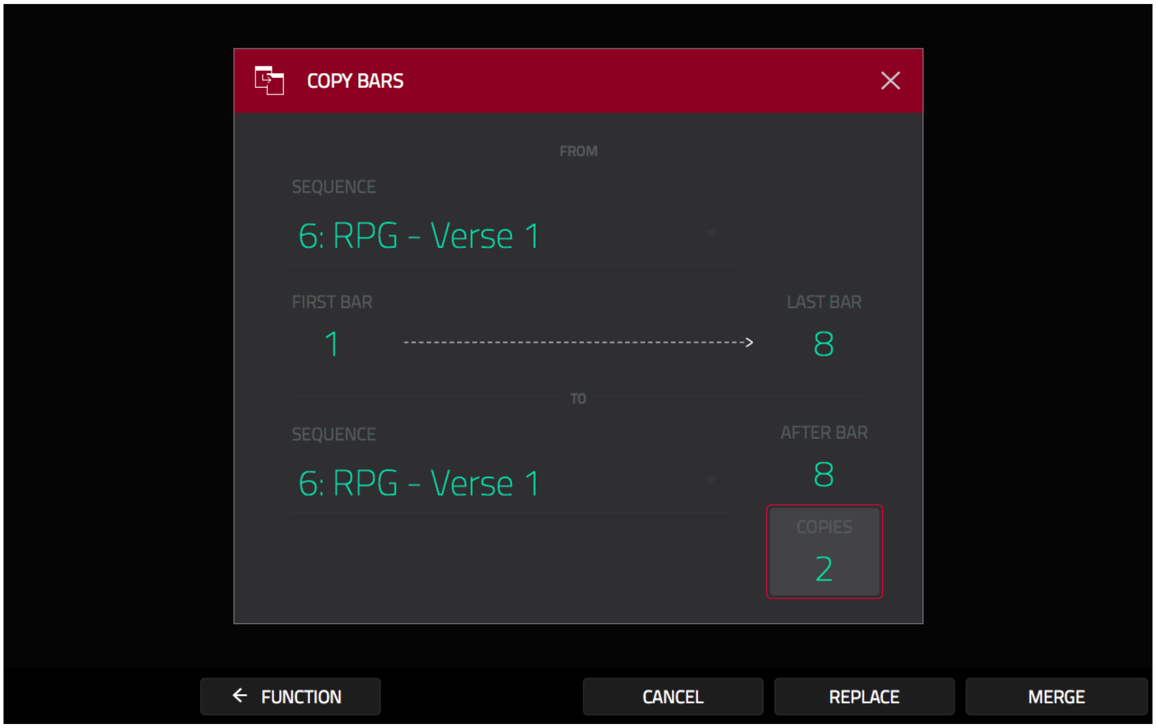
Check out my version on **sequence 5.**

Creating the First Verse

The plan for verse 1 was to have quite a long verse, and eventually record some vocals running over the entire length of the verse. As is often the case with a verse there may be some changes to the core structure of the instrumentation.

Go to **sequence 6** (**RPG – Verse 1**) and have a listen to the verse I created.

Once again, I used the core sequence template to build the structure of the verse. This time I used **COPY > BARS** to create another two copies of the core sequence, giving me a 24 bar sequence.



One thing that should be obvious is that the bass line is a bit different during the verse, in fact I just completely re-recorded it. Same key, scale and vibe, but slightly different melody and less repetitive.

There's also a couple of new sounds in the verse, so I've added two more tracks. From **bar 17** you'll hear a subtle pad sound (**track 9, 'Pad'**), and there's also a **reverse piano sound** on **track 10** that's been timed to exactly match the start of bar 9.

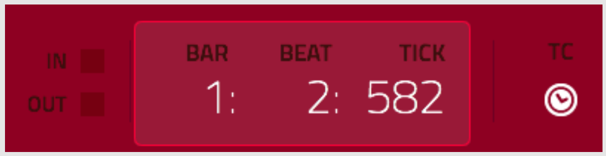
Finally notice that at the very end of the sequence, the guitar riff from the chorus comes in (track 9). This is because the guitar needs to come in for the chorus just before the start of the first bar, so if we want the guitar to come in at the beginning of the chorus, we need to introduce the sample at the end of the verse (this will make more sense when we piece the sequences together to make a SONG).

The Second Verse

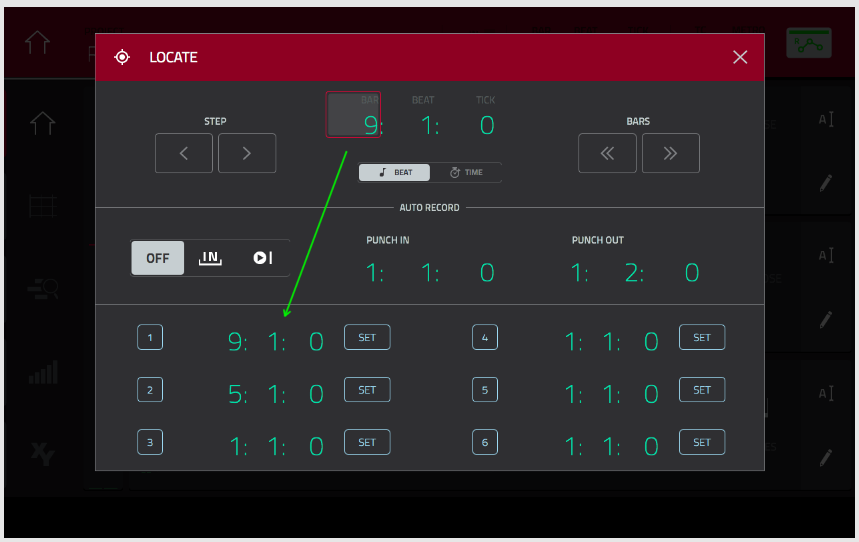
Verse 2 (**sequence 7**) is essentially verse 1, but this one is only 16 bars long. Have a listen and during the second half of the sequence it will quickly become apparent that I've introduced something completely different to the sequence. Say hello to **XYFX**.

Using the STEP and BAR Buttons

To navigate to a particular location in your sequence, you can use the **STEP** and **BAR** buttons. In the MPC X these are actual physical buttons above your transport control. In the MPC Live they are accessed on your screen. In **MAIN** double tap on the **time location display** in the top of the screen:



This brings up the **LOCATE** screen.



Tap the **BAR** buttons on the right to navigate in 'bar' increments. Tap the **STEP** buttons on the left to navigate in steps that match the current quantize setting.

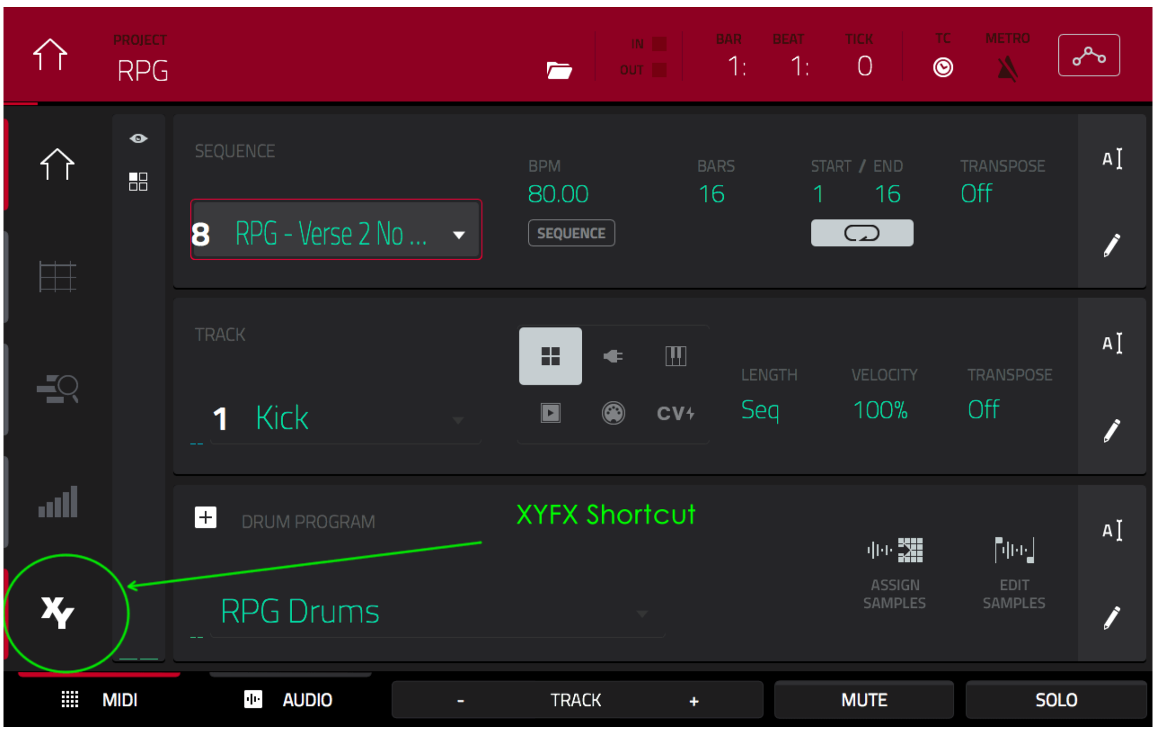
You can also set up to eight preset locations for your project – to set a shortcut to the start of bar 9, either enter **9: 1: 0** into shortcut '**1**', or navigate to **9: 1: 0** in the sequencer and press **SET** next to the shortcut number you wish to assign that time to.

Live Sweeps With XYFX

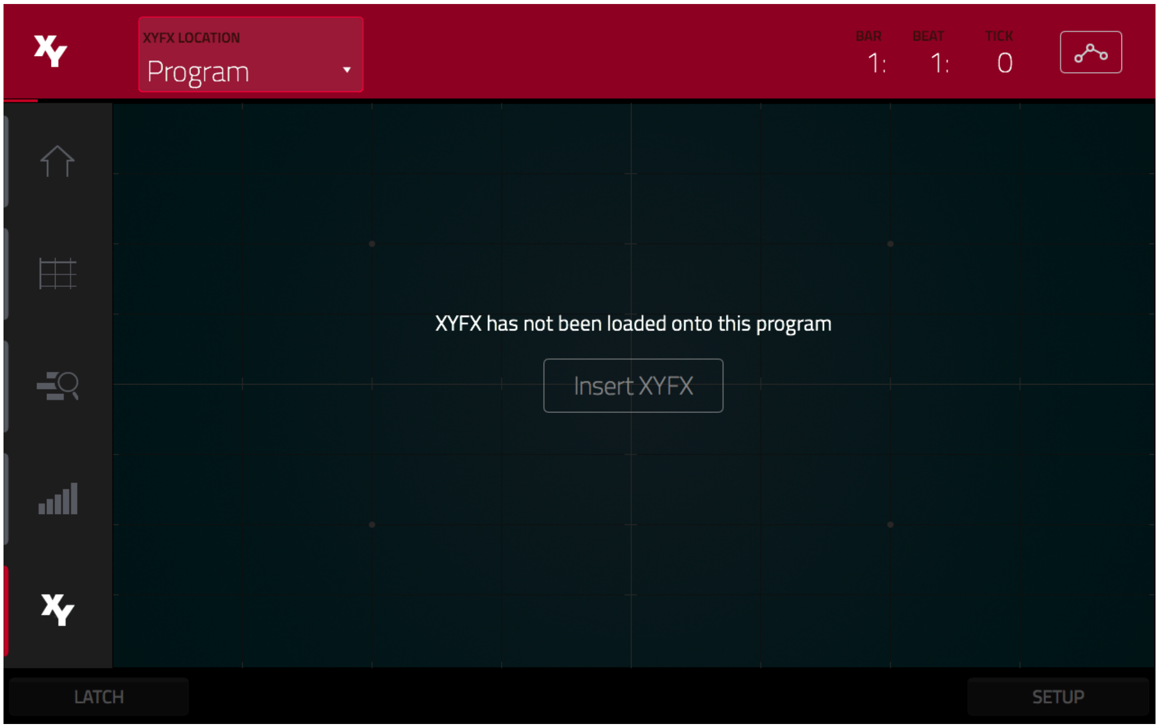
The MPC has a fantastic way of adding some cool live filtering to your sequences, all entirely from the touch interface.

From bar 9 onwards in verse 2 you'll hear that the drums in our sequence are heavily filtered, but as the sequence progresses the filter gradually opens more and more until it reaches a fully open pack at the end of the sequence; this was easily created using the **XYFX** mode.

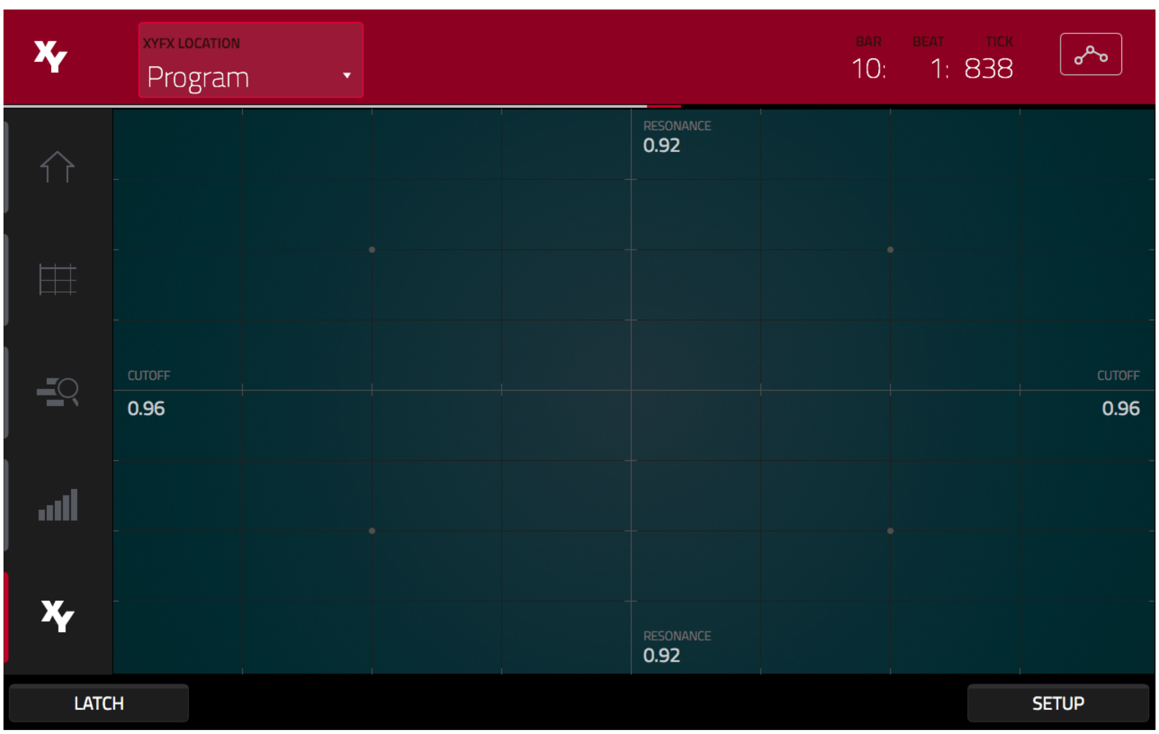
Go to **sequence 8**, which is a copy of verse 2 but without the XYFX filter changes. With track 1 selected, go to **XYFX** via the shortcut icon in **MAIN**:



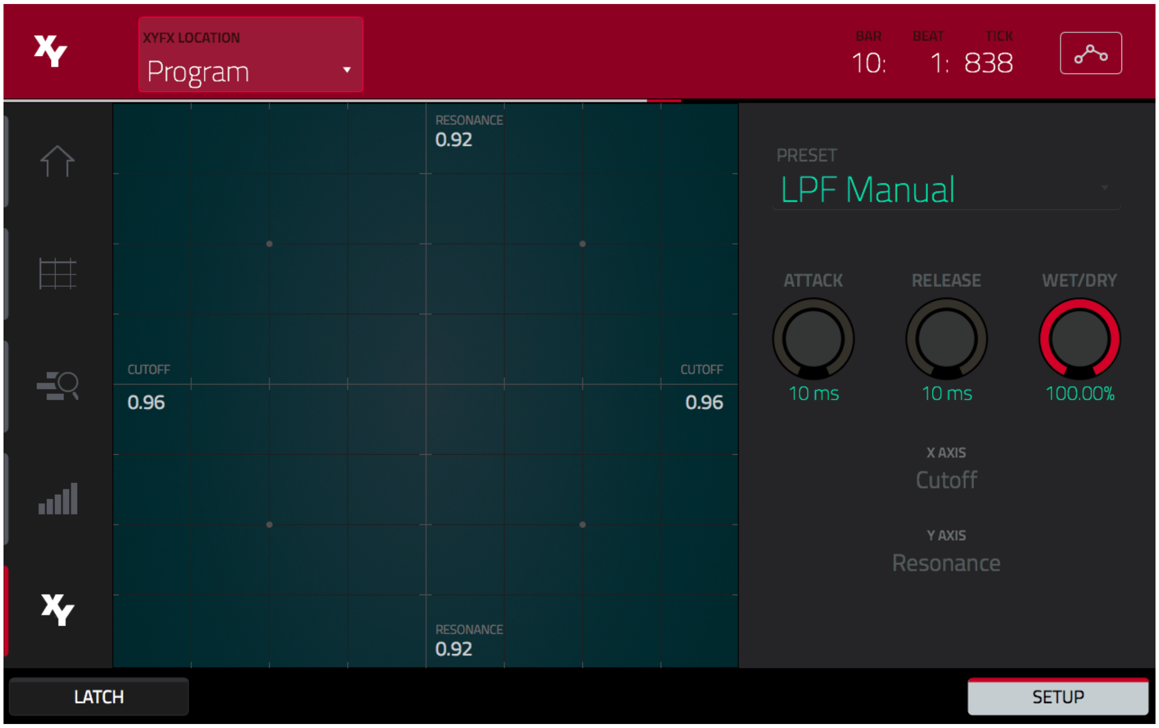
Alternatively you can get there via **MENU > XYFX**. If you've not activated XYFX on a program you'll initially see the following:



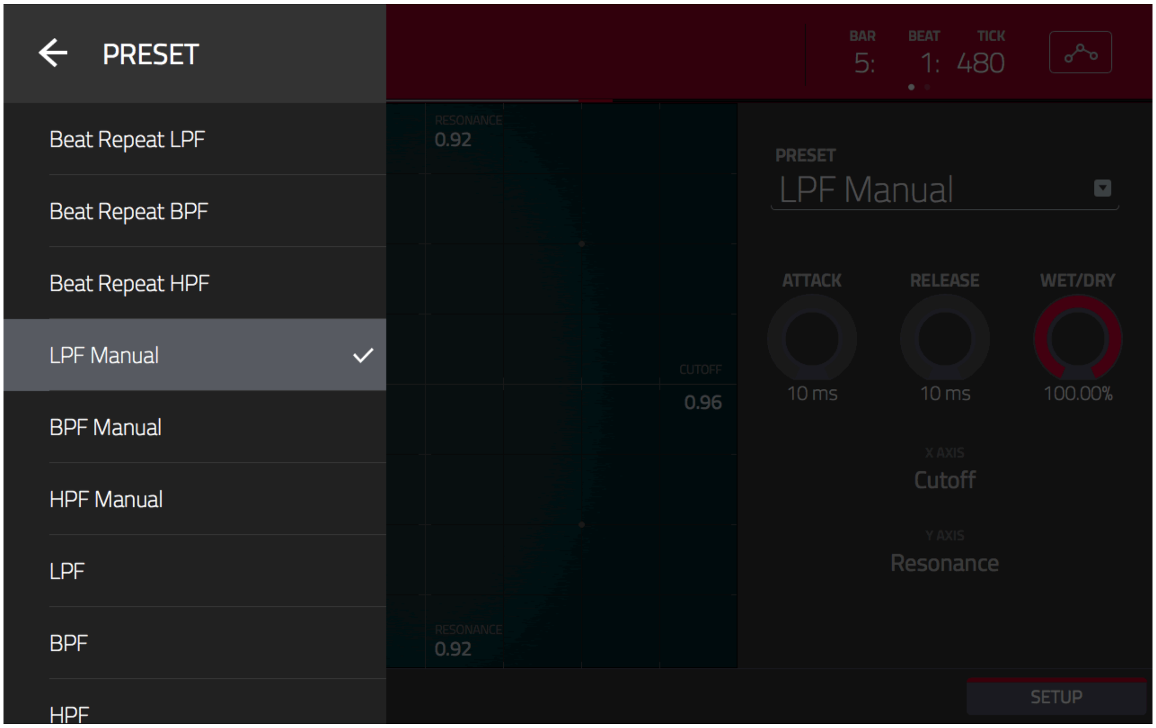
XYFX is basically an effect plugin, so it must actually first be inserted across your program (you'll need a spare program effect insert slot available). To load XYFX into a program, make sure **XYFX LOCATION** (top left of screen) is set to **PROGRAM**. Now simply tap '**Insert XYFX**'. You should now see something similar to the following:



XYFX mode lets you adjust filter settings across the program in real time simply by moving your finger around the touch screen. Hit the **SETUP** button:



In verse 2, I used the **LPF Manual** preset, but if you double tap the **PRESET** field you'll see there are many different filter options available:



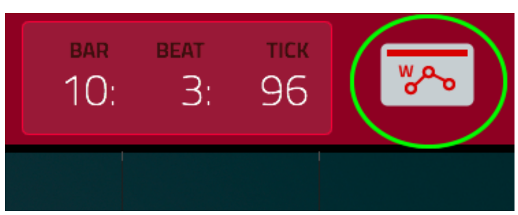
Back in the **SETTINGS** screen we can see that the **X AXIS** (i.e. going from left to right) is assigned to (filter) **cutoff**, while the **Y AXIS** (bottom to top) is assigned to (filter) **resonance**

The filter sweep I performed in verse 2 involved me starting with my finger pressed down in the bottom left corner of the screen and gradually dragging it diagonally to the top right corner. This therefore increased both cutoff and resonance values equally. It's important to note that XYFX is a program insert and therefore is applied to all the sounds in your program equally.

Hit **PLAY START** and with the sequence playing, try experimenting with XYFX to apply filter sweeps – remember you don't have to go diagonally, you could for example just change the filter cutoff by moving your finger from left to right across the middle of the screen. If you hit the **LATCH** button, XYFX settings will remain 'in force' even when you release your finger from the screen.

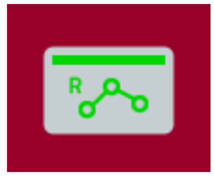
Recording Automation

To record XYFX performances to your sequence you must first ensure that **automation writing** is enabled. To do this, tap the automation icon in the top right of the screen twice so it turns red with a small 'w':



Once you are in write mode, hit **PLAY START** and your XYFX performance will be recorded to your sequence – with 'write' enabled, automation can be recorded without the need for 'OVERDUB'; your sequence just needs to be playing.

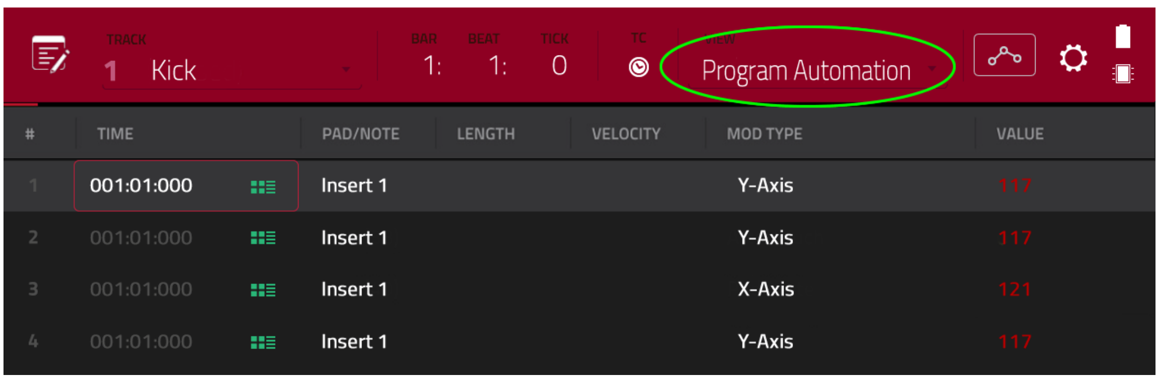
After recording you should change the automation setting to '**R**' (read only) by pressing the icon so it turns green:



Try experimenting with the different presets to see what other filtering effects you can produce!

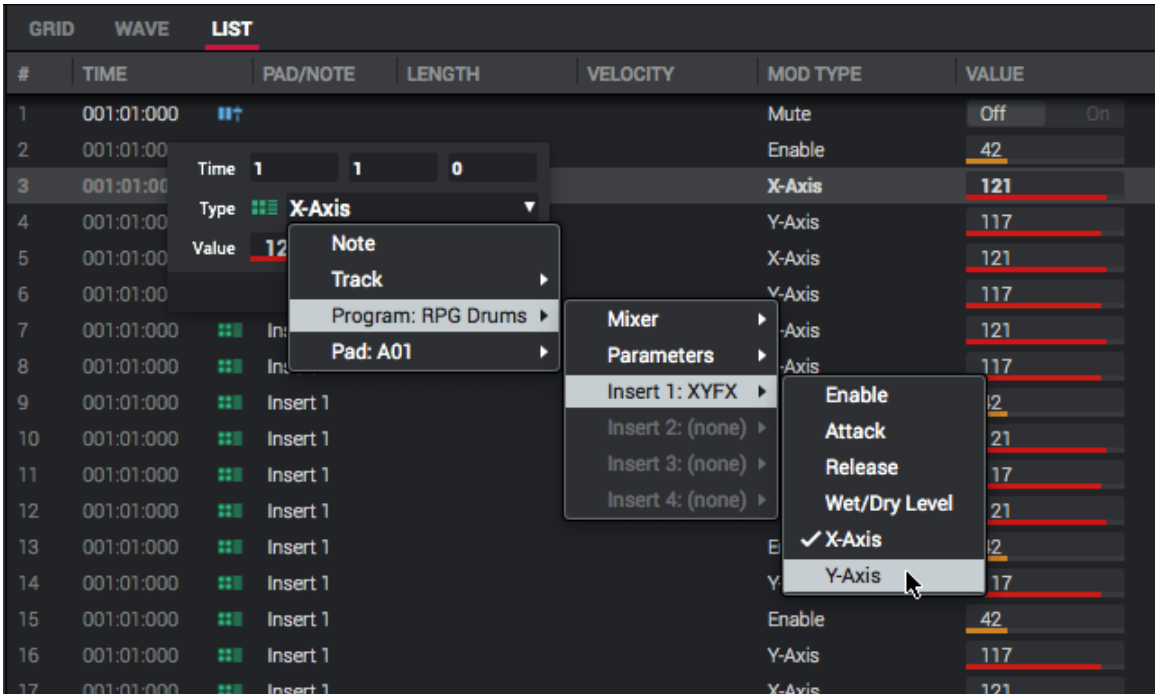
Editing Automation in LIST EDIT

You can view recorded automation events in LIST EDIT, for our XYFX data, make sure you select '**VIEW: Program Automation**' in the filter at the top of the page, as this will remove all other events from view:

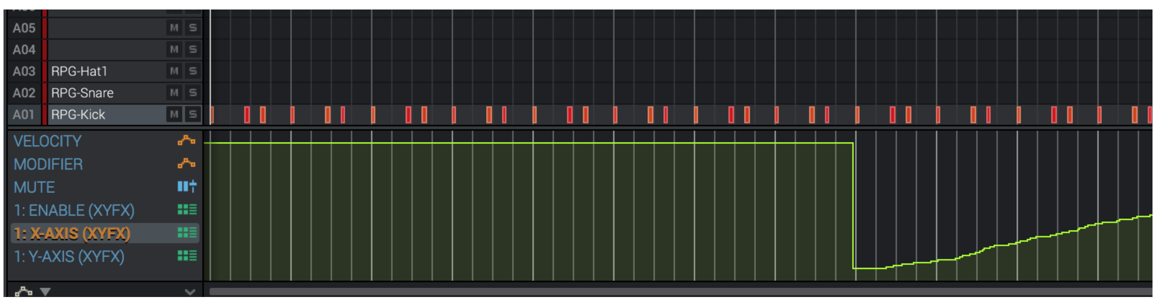


You can also edit this XYFX data - tap the 'value' for a specific event and change with the data wheel or double tap to bring up the number pad. Single tap an event and use NUDGE to move it around. However you cannot change any other XYFX parameter in the hardware UI.

However if you open your project in the computer GUI you can edit individual automation events very easily - just like note events, you can double tap any automation event and change all parameters.

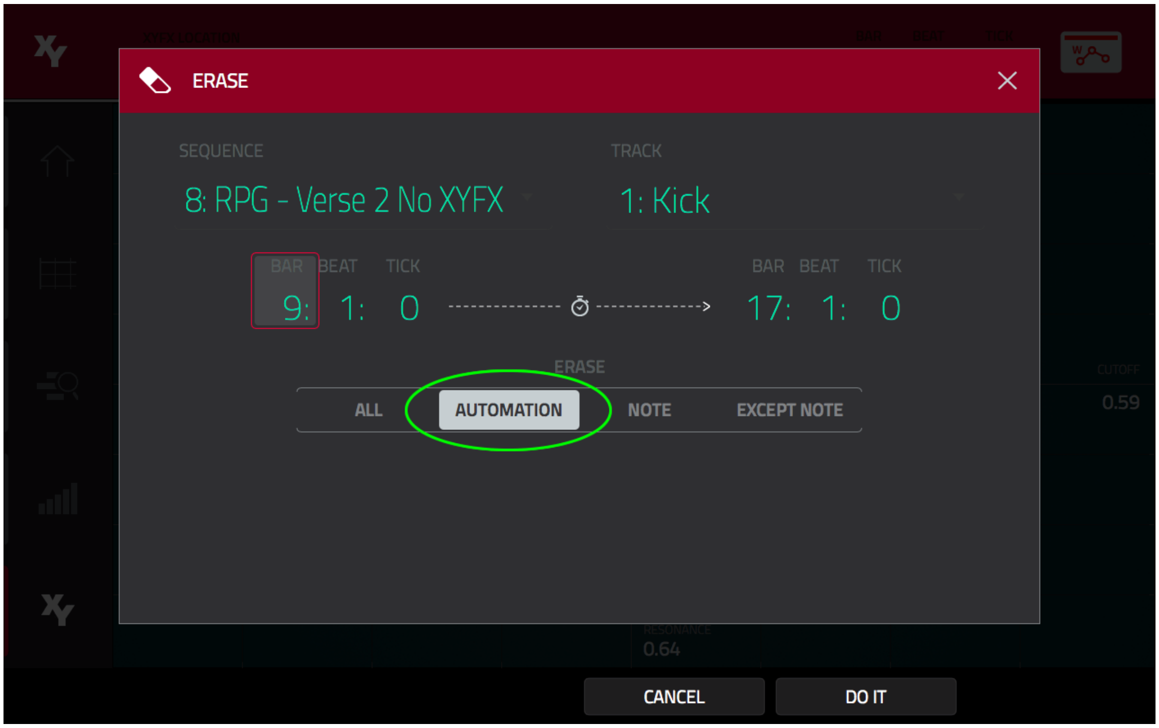


As well as graphically via the **GRID** '**modifier lane**':



From a hardware UI perspective I currently feel the best way to fix any errors in an XYFX performance is to just re-record it.

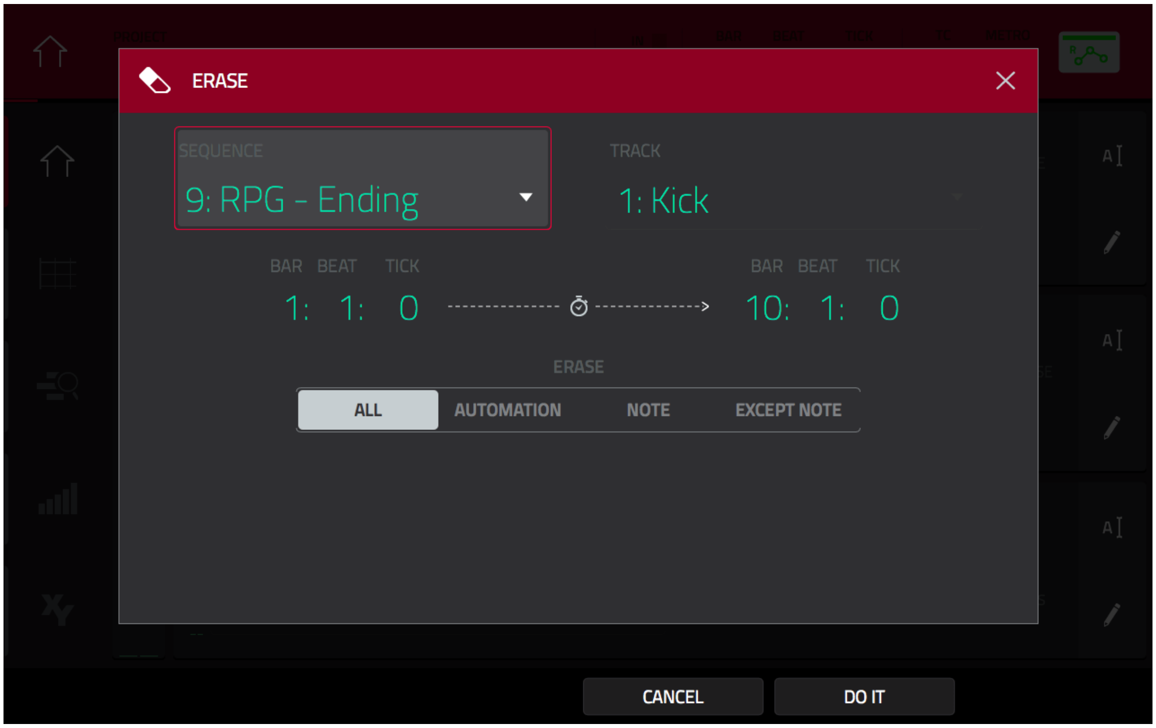
You can also use the **ERASE** function to delete all automation data within a specified time range. Just hit the **ERASE** button:



Select your **time range** and make sure the **AUTOMATION** tab is selected. This will however delete all automation events (as well as TRACK MUTES), there's currently no way to delete only one automation event type.

Creating the 'Ending'

Sequence 9 is the '**Ending**' which will finish things off for us. Here I've simply removed all the drums completely – this time rather than mute, I actually erased all the events. To quickly erase events on a track, hit the dedicated **ERASE** button (it's below the BANK D/H button).



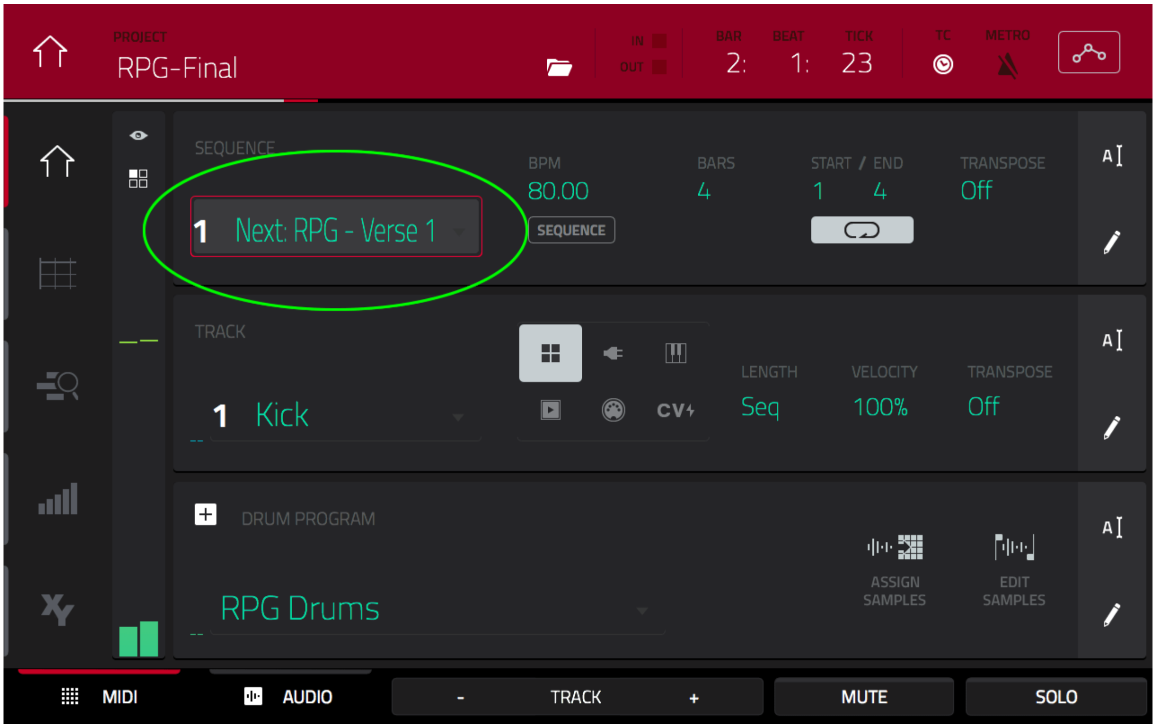
The default settings will simply erase all events off the entire track – hit **DO IT** and repeat for all drums.

Sequence Experimentation

We now have a bunch of sequences that will make up the structure of our song. So how do we chain all these sequences together and experiment with the running order? Well, we can either do it manually (as we'll see next) or we can permanently program our order of sequences into a song (which I'll cover later).

First, load up the '**RPG – Final**' project. Here I've just removed all the superfluous sequences, leaving us with our main sections – intro, verse 1, verse 2, chorus and ending.

As we have an 'intro' sequence, let's start with this first, so select sequence 1 (RPG - Intro). Press **PLAY START** so the sequence plays. In **MAIN**, tap the current sequence name and turn the data wheel one click clockwise (or press the + button). You should see the following:

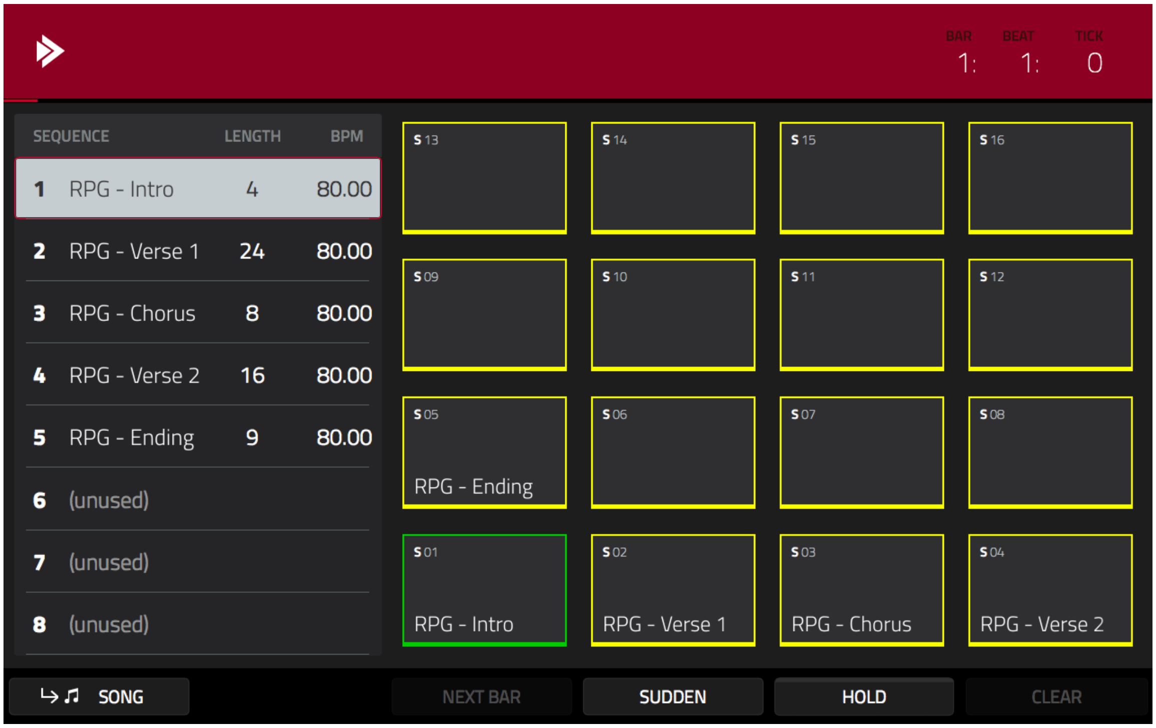


As we've seen earlier in the book, this indicates which sequence will play next; i.e. **RPG – Verse 1**. Leave the current sequence playing and when it reaches the end of the sequence, the MPC automatically plays RPG – Verse 1 for you, keeping everything perfectly in time. You could of course select any sequence to play here – simply keep on turning the data wheel or + button until you find the sequence you wish to play.

This method is okay when you are initially working on a song and have a few sequences you wish to mess around with, but it's not the best way to arrange a song in anticipation for recording or a live performance, nor does it lend itself to easy experimentation.

Next Sequence

When it comes to quick experimentation or using your MPC in a live situation, **NEXT SEQUENCE** mode is a great option. Go to **MENU > NEXT SEQUENCE**:



This screen allows you a lot more 'live' control over your sequence playback, and is very similar to the concept of the TRACK MUTE screen. Here instead of assigning tracks to each pad on the MPC, each pad represents *an entire sequence*.

The top left of the screen shows the currently selected sequence - when you press **PLAY START**, this is the sequence that will play first. Then while this is playing, press the pad that corresponds to the sequence you wish to play next – it will flash green, and in the bottom left you will be shown that it will play (once the currently playing sequence finishes). So with **sequence 1** (**RPG – Intro**) playing, hit pad **A02** to line up the **'RPG – Verse1'** sequence; once the intro finishes, the chorus will play automatically and in time.

With verse 1 playing, press pad **A03** to line up the chorus and this will begin playback as soon as verse 1 finishes. Carry on in this manner to experiment with the sequence structure until you find something that works well.

With the chorus playing, press pad **A04** to line up **verse 2** but then immediately press **NEXT BAR**. This time, the currently playing sequence carries on playing until it reaches the end of the *current bar* – at that point the next sequence will take over. Use this to override the current sequence length, so for example 'verse 1' is 24 bars long, you could use NEXT BAR to bring the chorus in after 8 bars.

The **CLEAR** button removes the sequence that has been pre-selected. This function is good if you decide that you don't want to change the currently playing sequence.

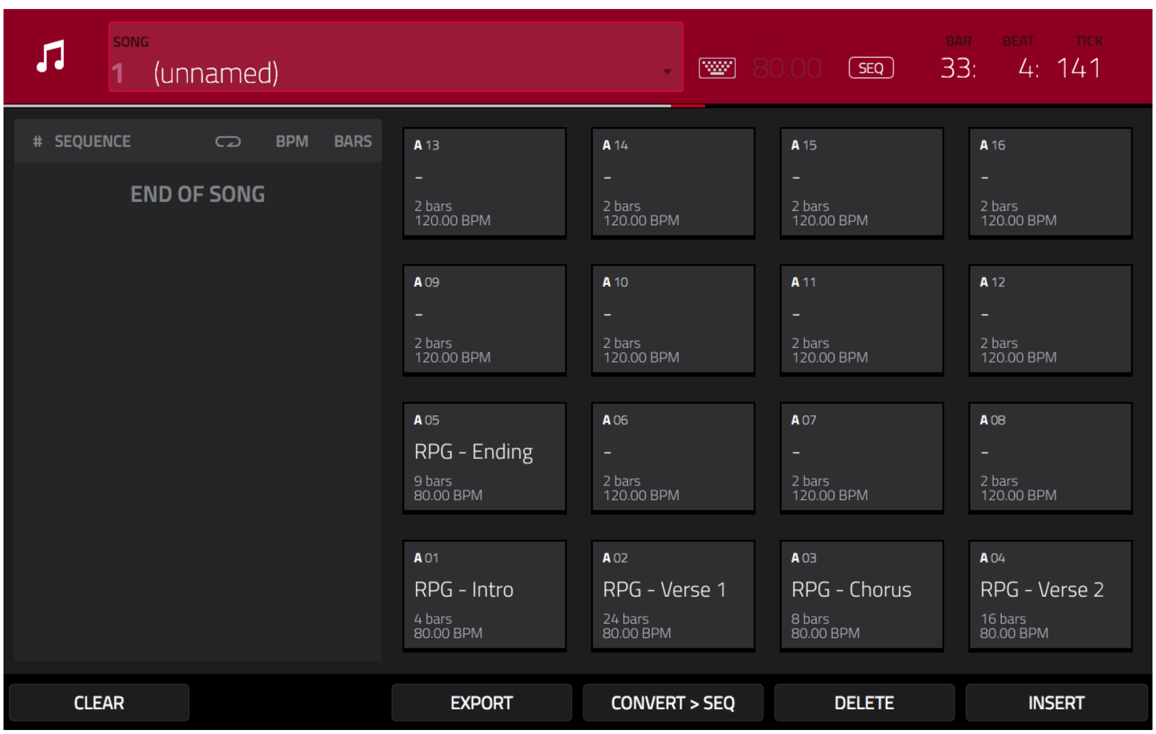
If you press **HOLD** while a sequence is playing, this will stop the MPC playing any lined up sequences, and instead it will just continue to play the current sequence. Hit **HOLD** again to release this next sequence (the next sequence will begin playback only once the current sequence is finished playing through).

Finally **SUDDEN** can be used to begin playback of a line-up sequence instantly. If you use this, make sure your timing is spot-on!

Song Mode

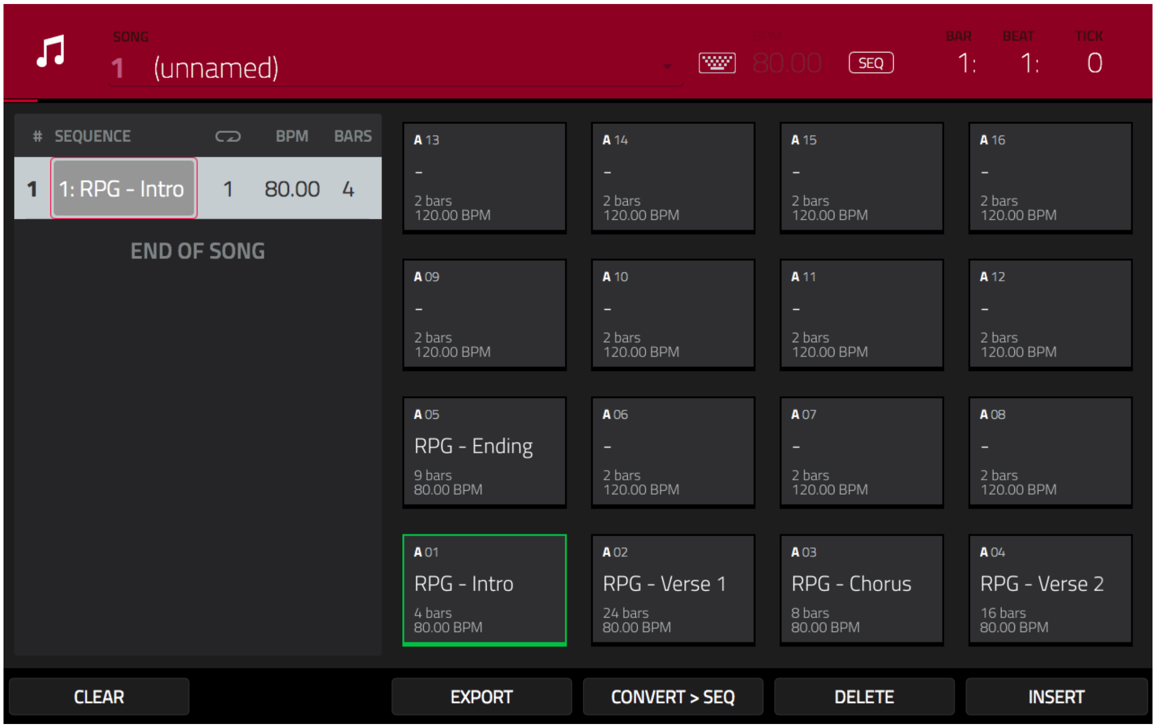
After playing around with your sequences using the above methods, you'll probably be looking to create a more permanent version of your sequence order. This is where **SONG** mode comes in. SONG mode allows you to join together multiple sequences to be played in a specific order, a specific number of times.

Go to **MENU > SONG**:



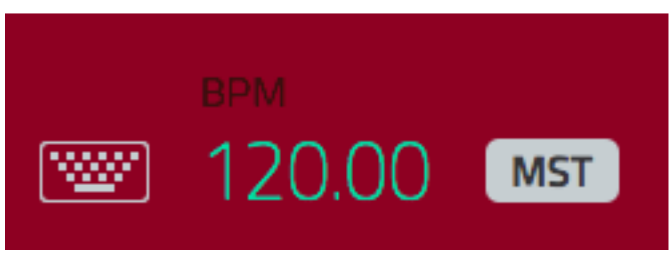
At the top is the name of your song (currently '**unnamed**'). On the left hand side you can set your sequence structure for this song, which is performed in 'steps'.

To add your first step, press the **REC** transport button. Now hit the pad that corresponds to the sequence you wish to add to the song playlist. Press pad **A01** to insert sequence 1 (**1:RPG – Intro**).



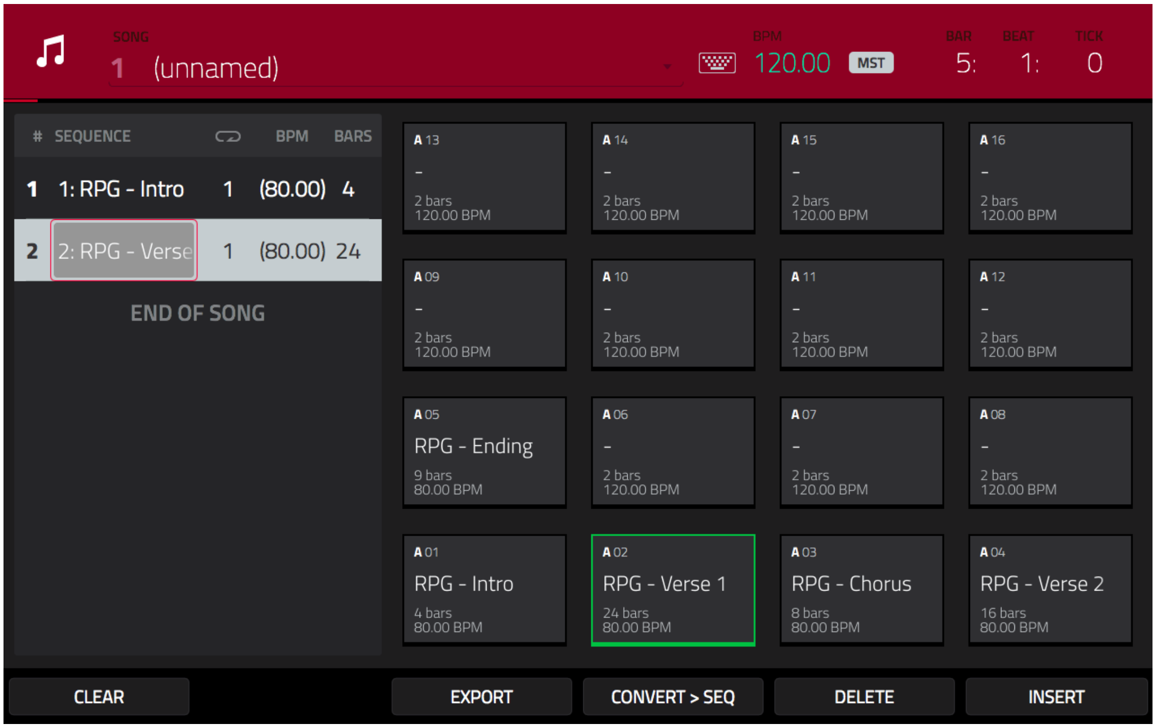
The default step setting will play the inserted sequence ‘1’ time – this field is editable and can be changed to any number of repetitions you wish.

The **BPM** field shows you the BPM of the sequence, as set in MAIN. If you have steps made up of sequences with different BPMs, each sequence will, by default, play at the BPM you set for that particular sequence in MAIN. However if you prefer to override all BPMs and instead have a single ‘master’ BPM for your song, tap on the **SEQ** icon in the top bar of the screen so it changes to MST:



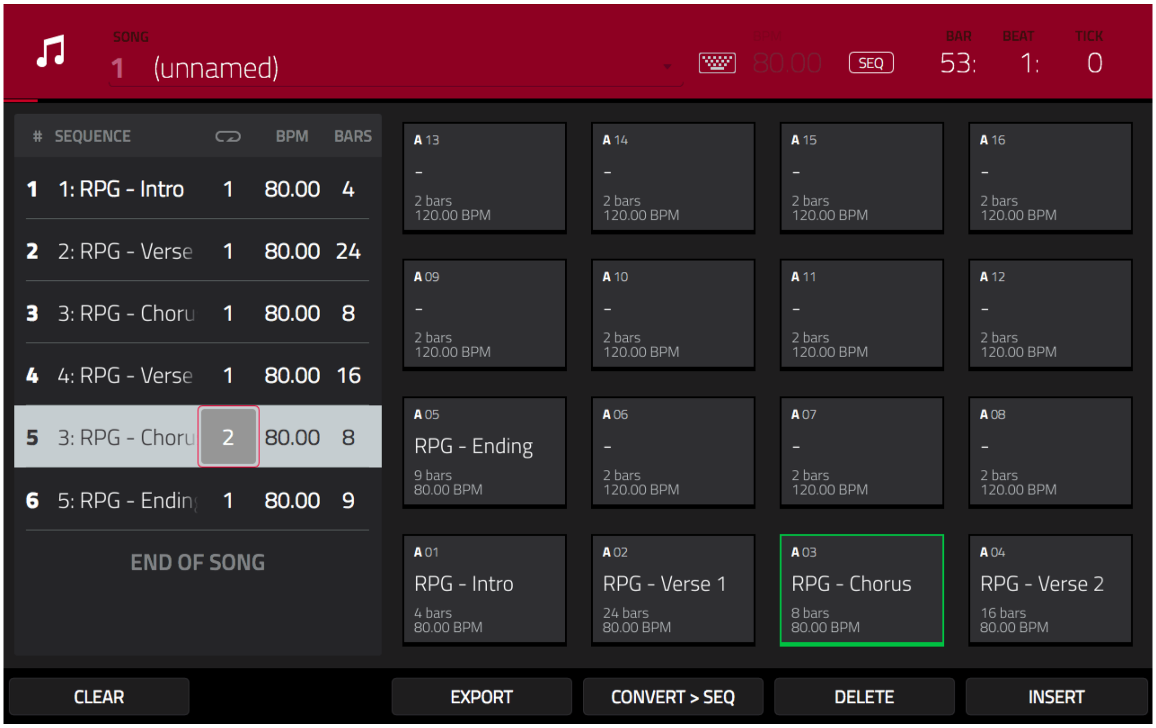
Then tap on **120.0** and change the BPM to whatever value you prefer.

You can now add further steps to your sequence by hitting the relevant pads; each new step is added immediately *after* the currently selected step. So press **pad A02** and the MPC will add **sequence 2** (**RPG – Verse**) to step 2.



Carry on building your song step-by-step using the following structure:

- Intro (1 repeat)
- Verse 1 (1 repeat)
- Chorus (1 repeat)
- Verse 2 (1 repeat)
- Chorus (2 repeats)
- Ending (1 repeat)



Notice how the chorus in step 5 is repeated twice before the 'Ending' sequence is played. You can see my version listed under **Song 2**.

Now hit **PLAY START** to hear the song.

025 Adding Vocals & Exporting Your Beats

Limitations of Song Mode

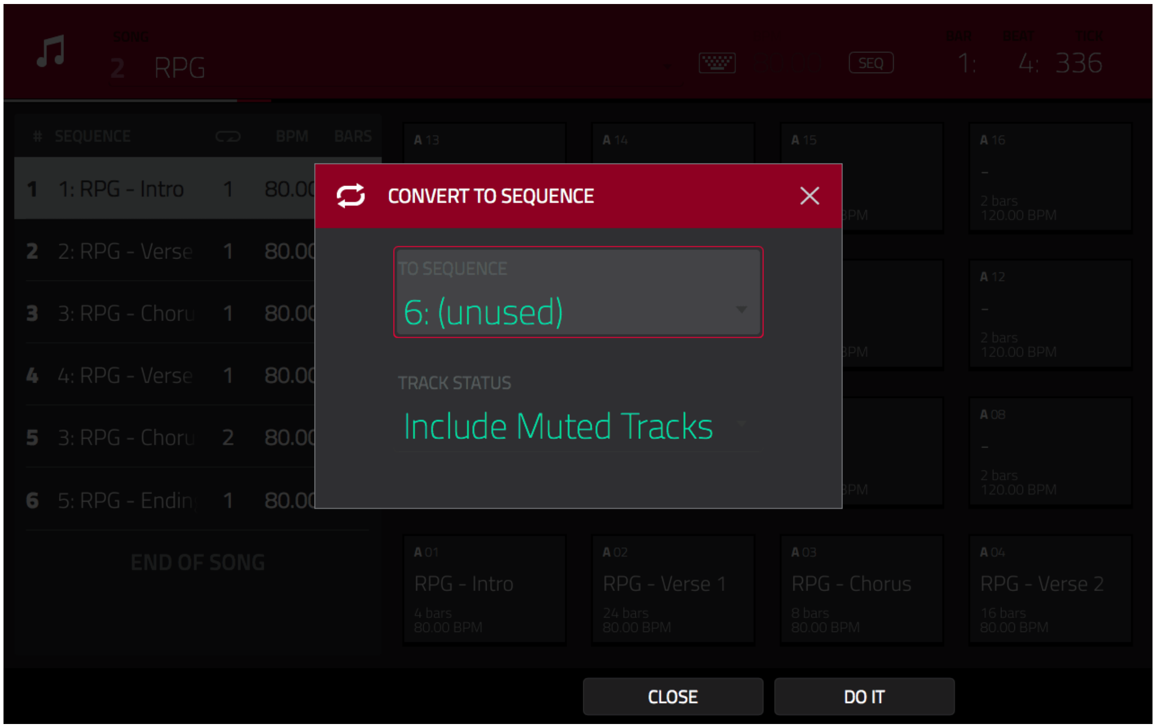
Song mode is a useful way of putting your various sequences together into a single composition and once you're happy with both the structure and content of your song, you're no doubt going to be ready to move on to the final stages of song creation.

For example, you'll probably want to head over to the **CHANNEL MIXER** to perform some tweaks to the mix; now for reasons unknown, going to the channel mixer take us out of SONG MODE, which isn't very useful as it means we're stuck mixing just a single sequence at a time.

Another issue with SONG MODE is that there's no way to now add any additional tracks to your song – for example, how do we record a vocal track across the entire 'song'? It's simply not possible in song mode, so what can we do?

Converting Your Song into a Sequence

The solution is to convert our song into a single sequence. Press the **CONVERT > SEQ** button at the bottom of the SONG MODE screen:



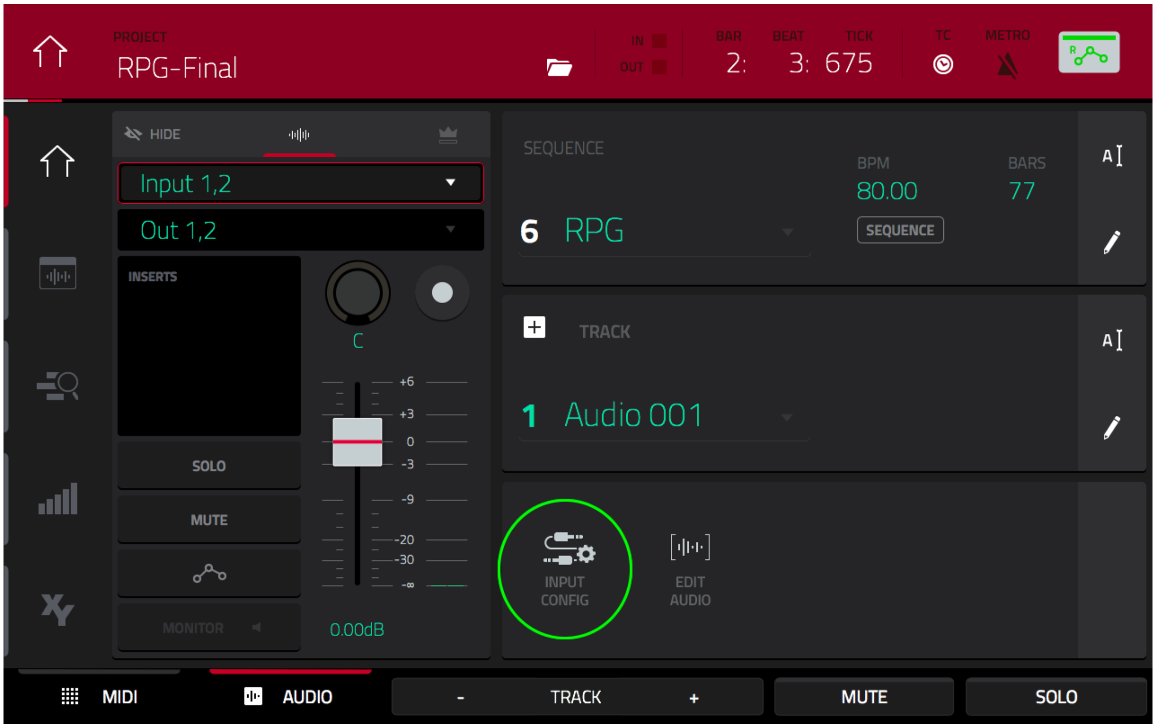
Select any blank sequence as your destination (‘**TO SEQUENCE**’), and for ‘**TRACK STATUS**’ set this to ‘**Include Muted Tracks**’ (you could set this to ‘Ignore Muted Tracks’ if there were muted tracks in all your sequences that are never used).

Hit **DO IT** to create your sequence. Now head over to **MAIN**, select this new sequence ‘**From Song 2 – RPG**’; give it a name ‘**RPG**’ and hit **PLAY START**.

With our song converted into a single, 77 bar sequence we can now not only mix our entire song using the various CHANNEL MIXERS, but we can also very easily record a vocal across the entire song.

Recording a Vocal Over Your Song

You can record a vocal over your audio track from MAIN or from TRACK VIEW. First, head over to **MAIN** and select **AUDIO** view. Click on **INPUT CONFIG**:



Here you can configure which recording input(s) you wish to record your vocal through. The default recording inputs are 1 and 2 (**Input 1,2**); vocals are typically recorded with a mono mic so if that's the case set this to a single input channel, such as **Input 1**.

Recording with Microphones

There are generally two types of microphone used for recording vocals; **dynamic** and **condenser**.

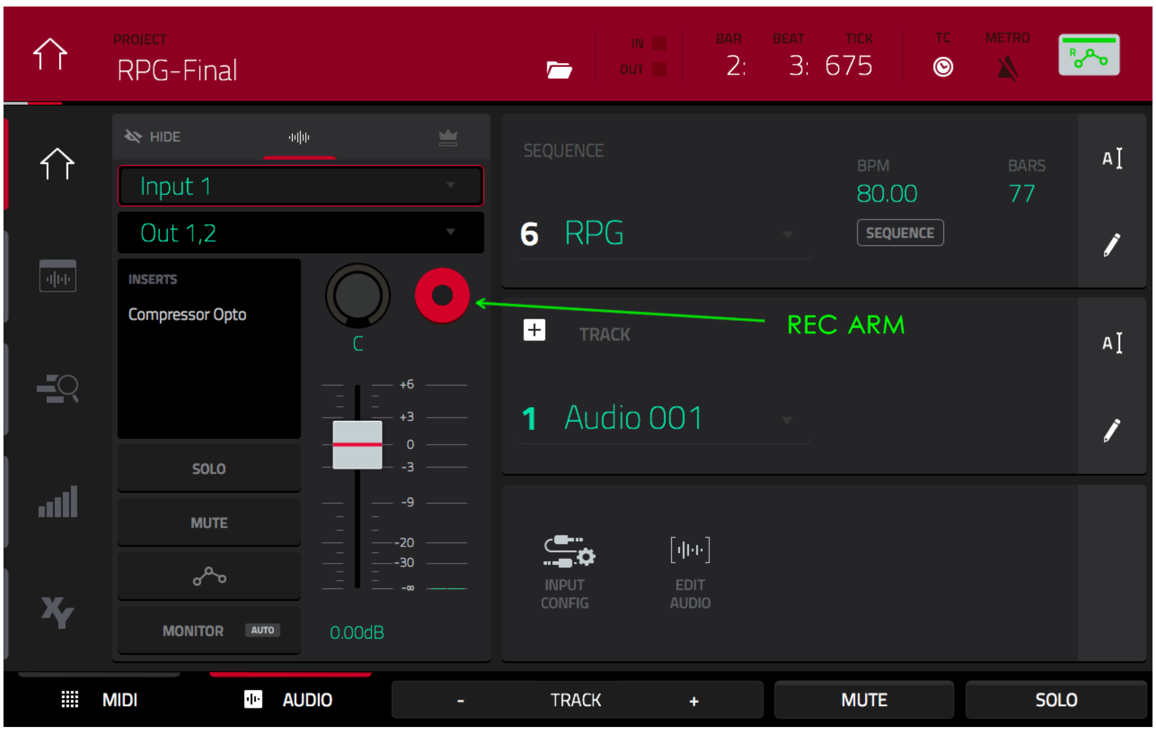
A dynamic mic can be plugged directly into a recording input of your MPC, while a condenser will also need a 'phantom power' supply. The MPC X has built-in phantom power (turned on using the **+48v** button in between the GAIN 1 and GAIN 2 dials), the MPC Live will require a condenser mic is fed via a suitable 'mic preamp' (either a dedicated box or often built in to mixer).

Tap on the **MONITOR** button until it reads **MONITOR AUTO** – you'll now hear the incoming vocal performance (but only when the track is armed).

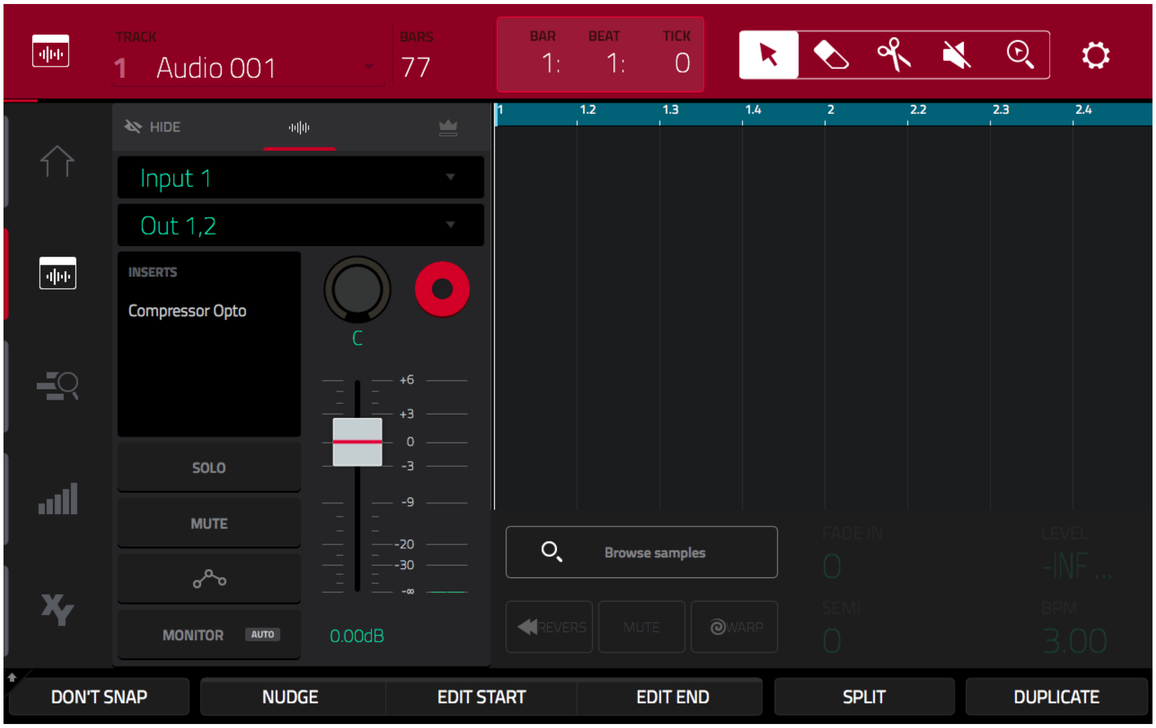
You can now adjust your input levels to ensure you get a good incoming level. If you wish, you can also apply an effect while recording the vocal by clicking in the familiar **INSERTS** box. This is best suited to 'dynamic' inserts such as compression as this can help reduce any excessive peaks in the incoming vocal performance which might lead to a distorted recording.

Just remember to take it easy with the compression as you don't want to ruin the performance, so if in doubt use less compression (or none at all) and instead ease back on the GAIN dials so the vocal stays comfortably in the 'green' portion on the meter. Get your singer to sing naturally including practicing some of the louder parts of the performance so you can gauge the most suitable GAIN levels.

Tap on the **REC ARM** button so it turns red:



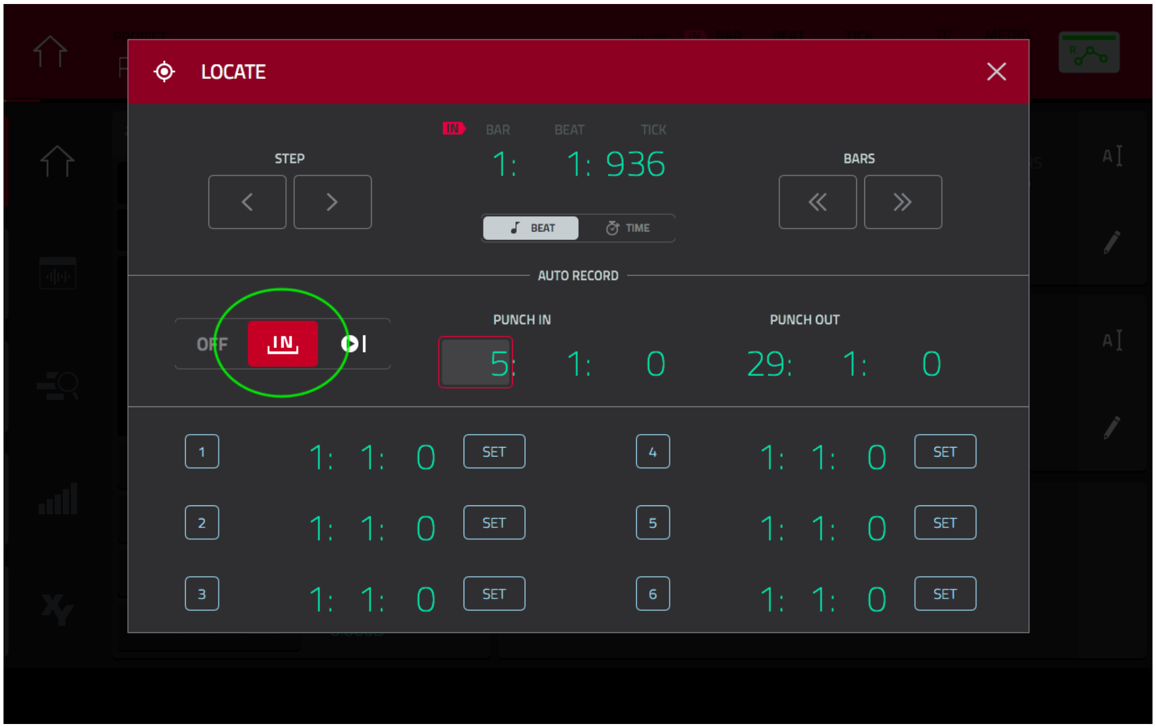
Hit the **AUDIO EDIT** icon, this way as you record you'll see your audio track appear in real time on the right hand side of the screen.



Press the **REC** button. To record from the very beginning of the song you'll just need to hit **PLAY START** (you'll get a count in if you've enabled that in MAIN's metronome settings).

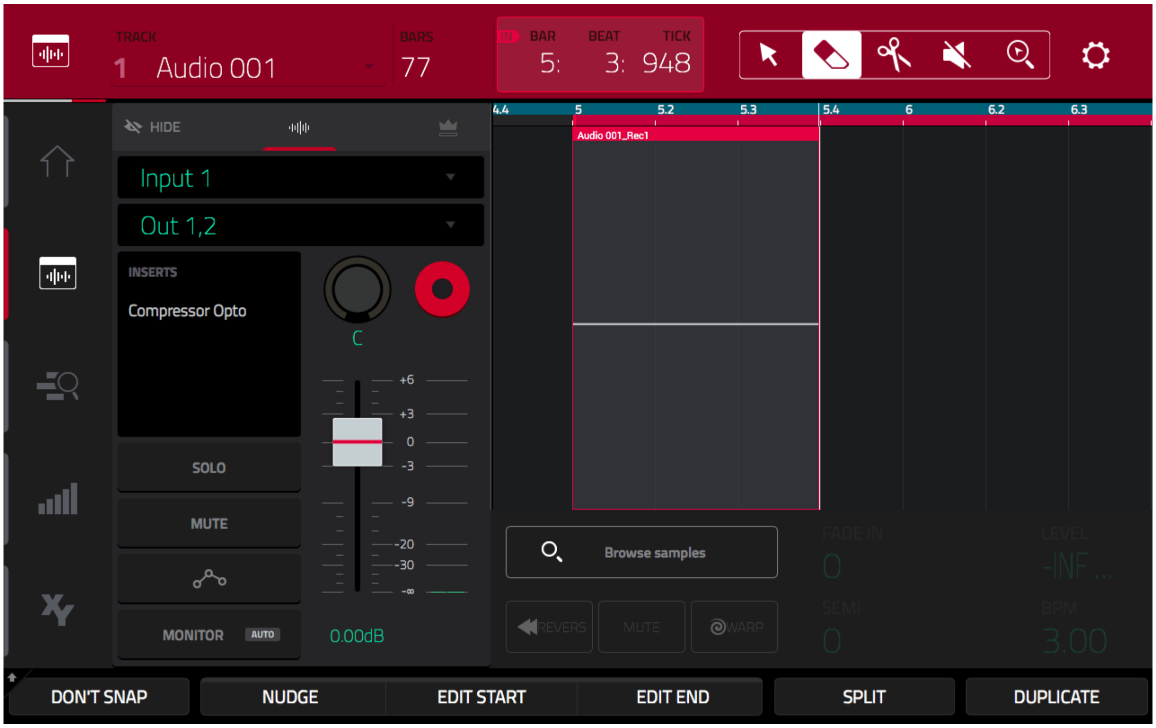
Alternatively, you can choose to begin the recording from any time location; let's assume that you just want to record the vocals for the first verse, which runs from the start of **bar 5** to the end of **bar 28**. One option is to set your **time location** to **5: 1: 000** and then press **REC** then **PLAY** (not PLAY START) to instigate the recording from the start of bar 5.

You can also set a **PUNCH IN** and **PUNCH OUT** range. Double tap the time location indicator at the top of the screen (or the **LOCATE** button on the MPC X):

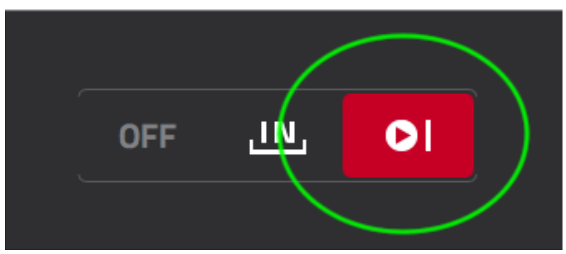


Under the **AUTO RECORD** settings, set the 'auto record selector' to '**IN**'; set the **PUNCH OUT** time first to **29: 1: 0** and then the **PUNCH IN** time to **5: 1: 0**.

Now you can press **REC** and **PLAY START**, and the sequence will play from the beginning, but it will not start recording until the sequence reaches the start of bar 5 and it will automatically stop recording at the end of bar 28:



Another **AUTO RECORD** option is to use **RECORD START**:



With this enabled, the entire sequence will play through without recording from start to finish before playing back a second time with recording enabled. Good for recording over shorter sequences.

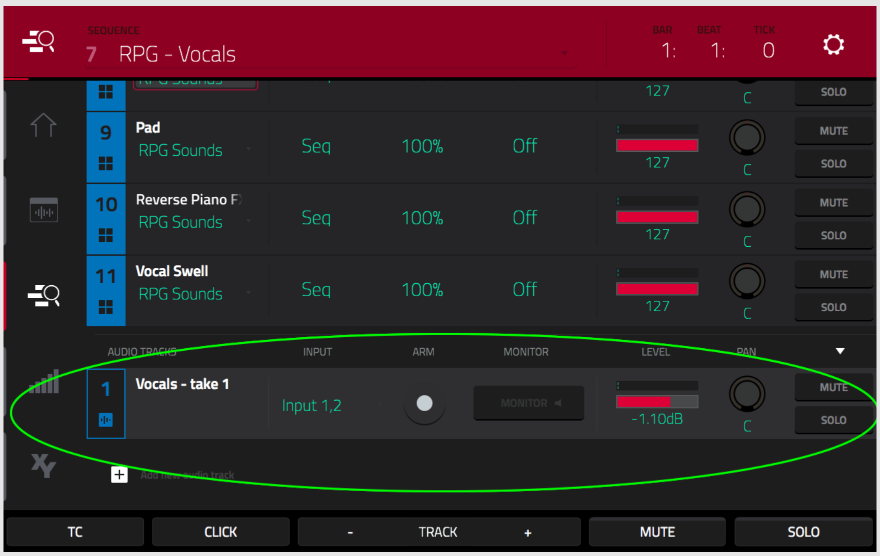
Recording Vocal Takes

There no mechanism specifically built in to handle recording additional vocal ‘takes’, so the way I’d suggest handling this is to simply record each set of takes to their own unique tracks and mute the ones not required – just remember that each vocal track will use up a fair chunk of memory and CPU, so don’t go overboard with this and remove from the project the ones you definitely don’t need.

Load the project file **RPG-Final** and go to **sequence 7** (**RPG – Vocals**) where I've recorded a vocal track to audio track 1 covering verse 1 and verse 2 specifically.

Recording Audio Tracks Via TRACK VIEW

You can also record audio tracks from the **TRACK VIEW** screen:



The only thing you cannot do from this screen is insert effects across the incoming signal. Please note that although you can manage multiple audio tracks in this screen, it is currently not possible to record multiple audio tracks *simultaneously*.

Performing a Rough Mix of Your Song

With your song now available as a single sequence, you can head over to **MENU > CHANNEL MIXER** and hitting **PLAY START** will begin playback of your entire song, allowing you to begin tweaking your mix in real time.

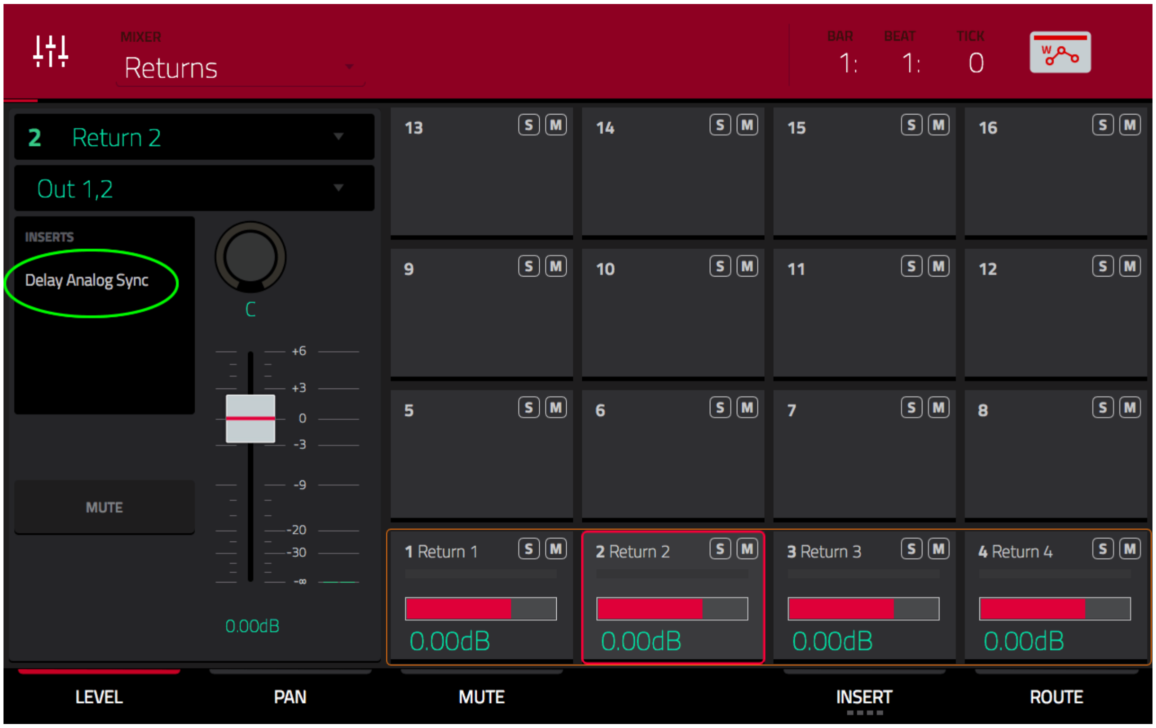
We came across all the various mixing options back in **chapter 013**. Each program has its own 'PAD MIXER' and insert effects, and much of the mixing work in the current song is already initially set up through that. For example, the guitar sample on **track 8** uses a '**Delay Mono**' effect as a pad insert on pad A01 in the **RPG – Sounds** program.

Our drums in the program '**RPG - Drums'** are already 'internally' mixed in terms of levels and panning using that program's own pad mixer. I've also applied an **MPC3000 vintage effect** across it via the **CHANNEL MIXER > Programs > INSERT**.

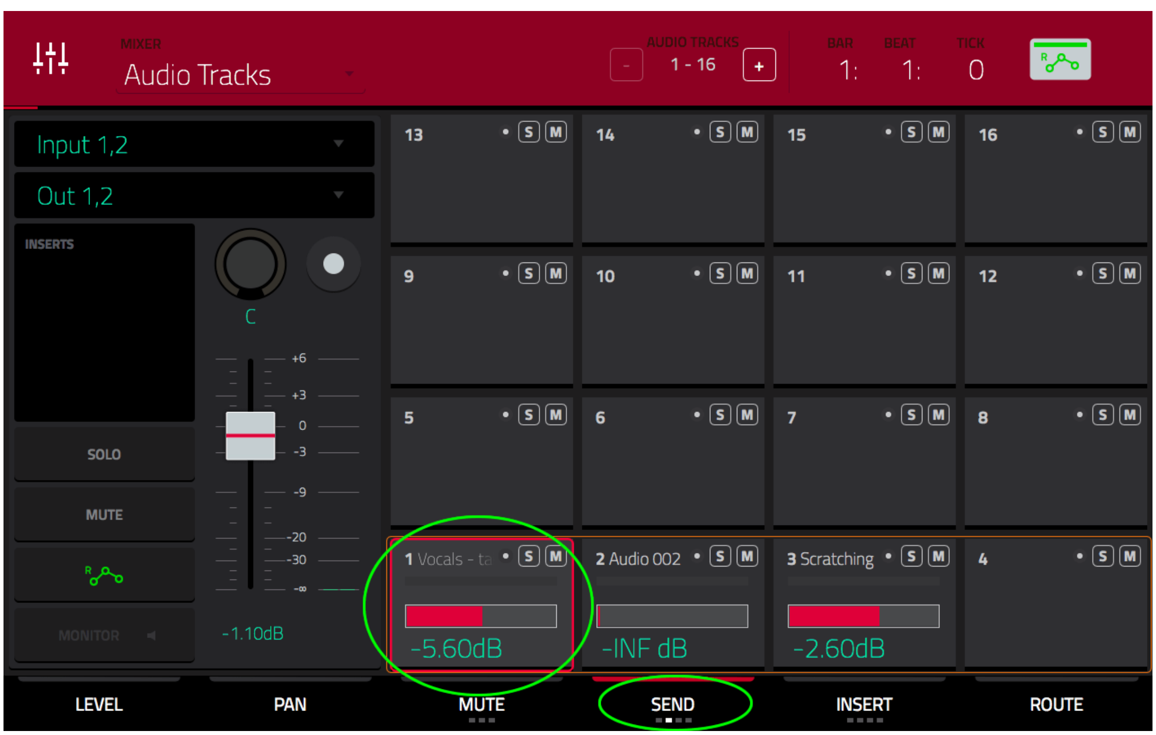
In **CHANNEL MIXER** mode, go to the '**Returns**' page and tap on **Return 1**.

Here I've assigned a '**Reverb Medium**' effect which can be used to add reverb to any channel we wish via the 'SEND' system.

Click on **Return 2** and you'll see I've assigned a **Delay Analog Sync** effect to this.



Select the **Audio Tracks** mixing channel and press the **SEND button twice** to show the send levels for **SEND 2** on the vocals track:



As you can see, I've increased the **SEND** value from **'–inf'** to **-5.60dB**, giving the vocals a subtle delay (I prefer using delay on vocals as I find reverb can make them sound 'muddy').

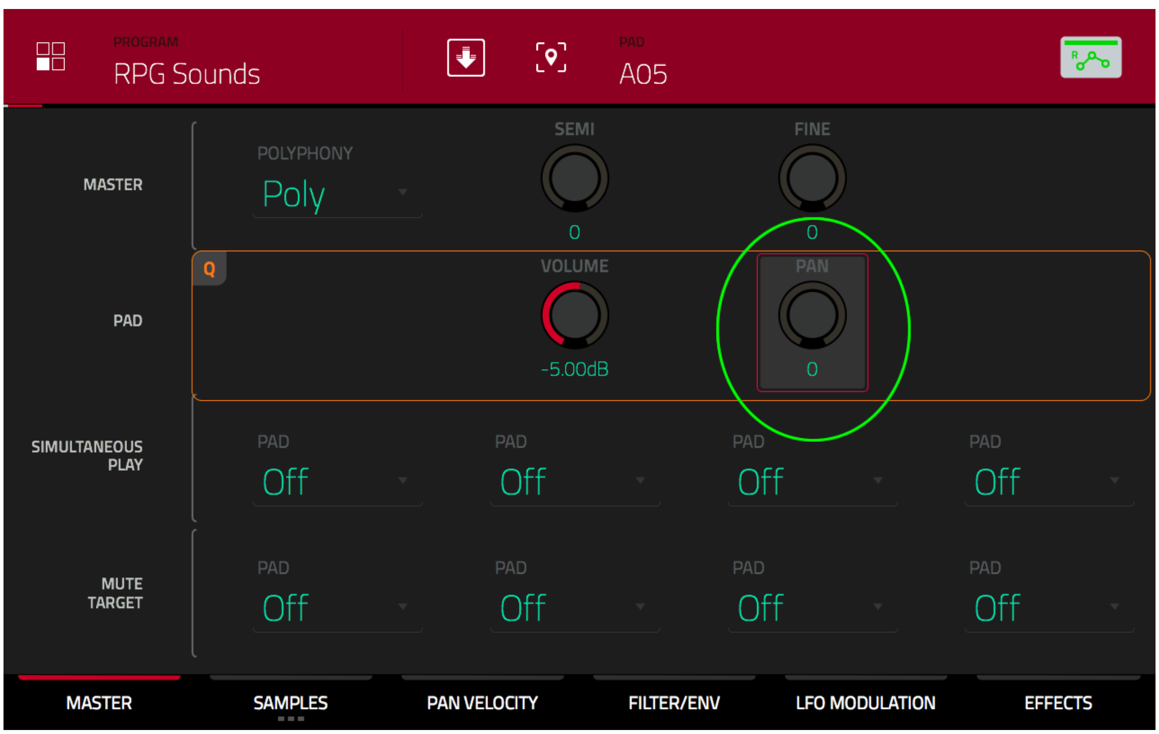
I've also add some reverb in the **CHANNEL MIXER > Programs > SEND 1**' to the **RPG Drums** and '**Old Piano'** programs.

Finally go to the **MASTERS** channel to see I've applied an **MPC60 Vintage effect** and a parametric EQ (**PEQ 2-Band**) across the entire output of the song.

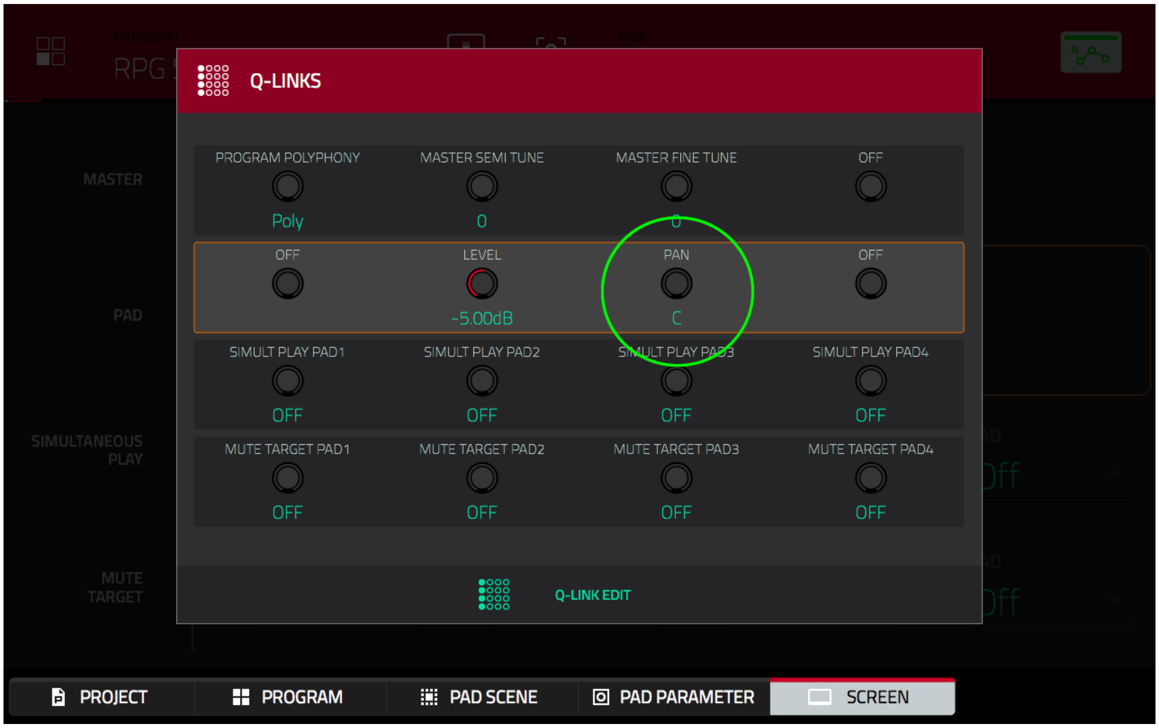
Recording Automation With the Q-Links

If you select **track 11** (**Vocal Swell**) I added a sample of the singer saying 'rise' which has been pulled from one of the vocal takes and mangled a bit. It's the '**RPG-vocal-swell**', assigned to pad **A05** on the **RPG Sounds** program. Let's add some additional automation to this to make it a bit more interesting; a pan sweep and a filter sweep.

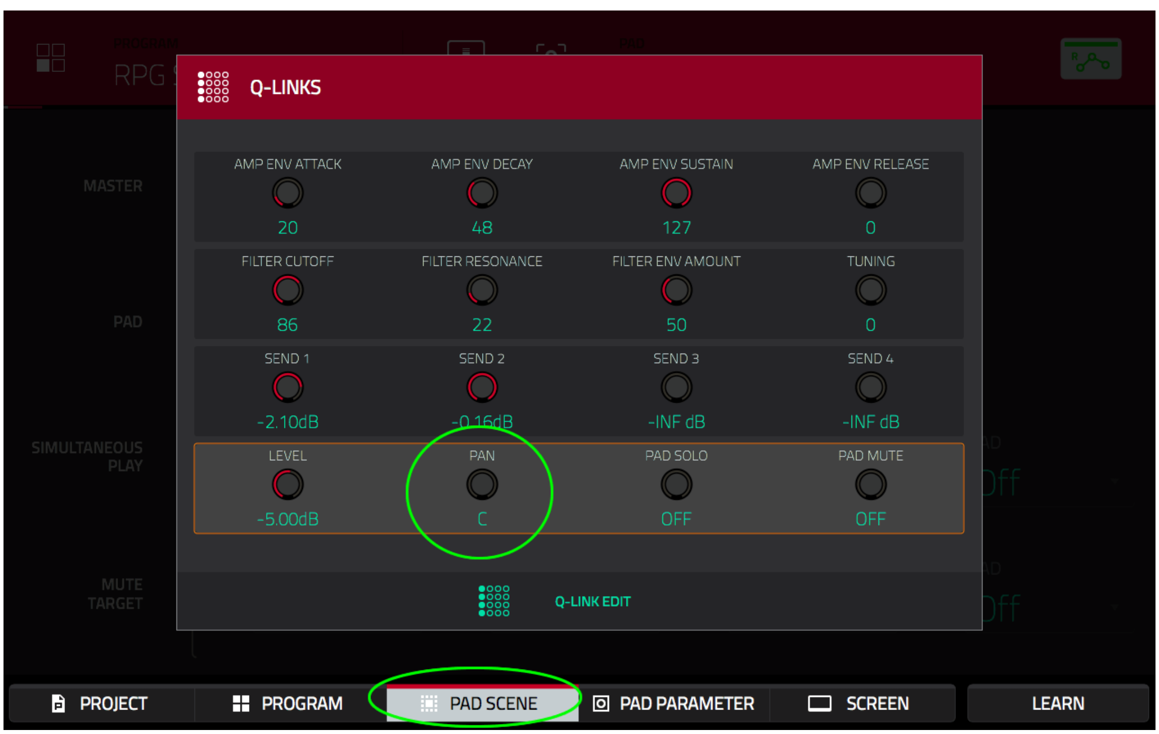
Now one way to perform a pan sweep is to head over to **PROGRAM EDIT > MASTER**, where we have a dedicated **PAN** control for the currently selected pad.



You can change the PAN in real time by tapping the PAN parameter and using the data wheel to change the value while the sequence plays back. Or you could use the Q-LINKS; as we can see from the Q-LINK page, this is done via **Q-LINK 11** (the **2nd Q-LINK from the bottom, column 2**).



Now, the Q-LINK assignments you see by default are referred to as the **SCREEN** assignments; the assignments change with each screen you visit. While in the QLINK page, tap on the **PAD SCENE** button (MPC X owners can just hit the dedicated **PAD SCENE** button):



With PAD SCENE selected, the Q-LINKS are mapped to a selected number of program parameters for the currently selected pad, such as envelope and filter parameters, send levels, tuning and panning. This is a 'global' Q-LINK configuration; it doesn't matter what page you are currently in, the Q-LINKS remain fixed at these assignments.

Q-LINK Options

In addition to PAD SCENE there are other pre-configured Q-LINK set ups available as well; PROJECT, PROGRAM and PAD PARAMETER each feature pre-configured Q-LINK assignments that remain 'globally' in place no matter which screen you are currently viewing. Each one can be customised to suit.

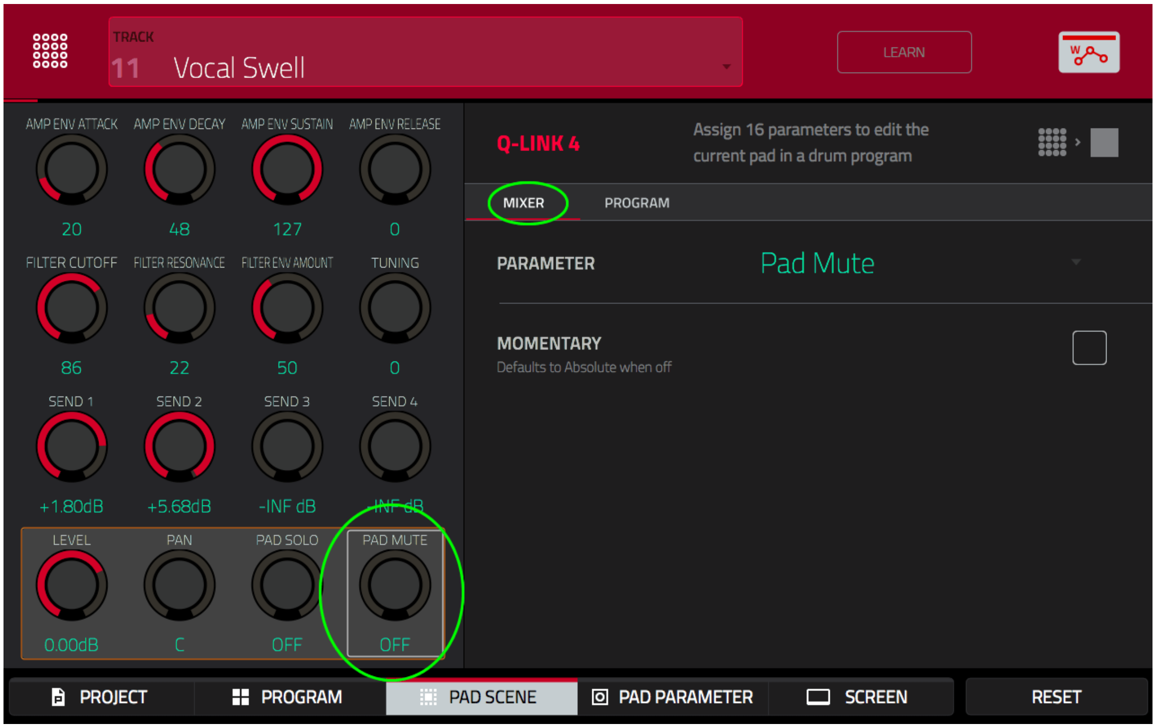
These pre-configured Q-LINK set ups are ideal for live performers who need very specific Q-LINK configurations.

If I wanted to record a 'pan sweep' using the PAD SCENE assignments, I would use the **2nd Q-LINK from the top, Q-LINK column 4.** I can do this from any page, not just from PROGRAM EDIT.

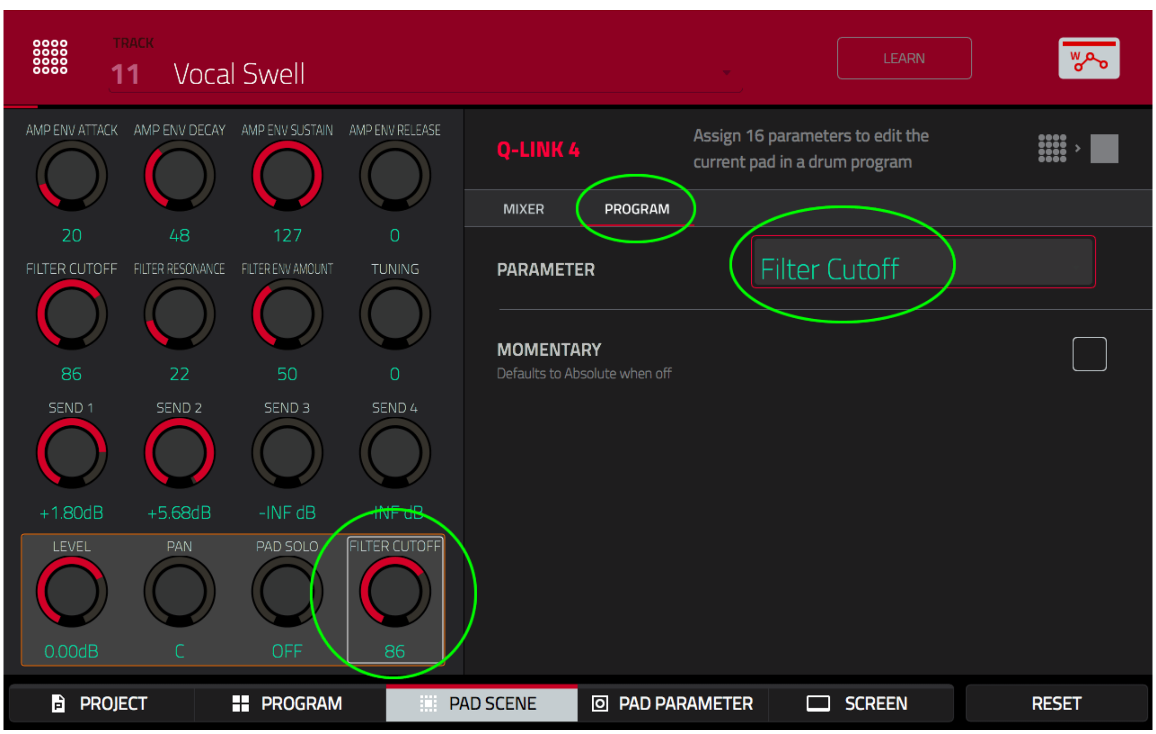
And how about performing a filter sweep at the same time as the pan sweep? Well **FILTER CUTOFF** is also on this page, **column 2, top Q-LINK**. So we could adjust both Q-LINKS simultaneously – but only in the MPCX as in the Live, different 'columns' cannot be accessed simultaneously.

So in the Live we can either record each sweep separately, but we can also re-configure the PAD SCENE page to suit our specific needs.

Go to **MENU > Q-LINK EDIT**, select **PAD SCENE** and tap **PAD MUTE** > **QLINK 4** (the **bottom Q-LINK, column 4**):



On the right of the screen you can re-configure the parameter this particular Q-LINK will control. Above the parameter you can see it is currently set to **MIXER** – tap the **PROGRAM** option next to it. Now set the **PARAMETER** value to read **Filter Cutoff**:



So now even with a Live we can control these two parameters at the same time.

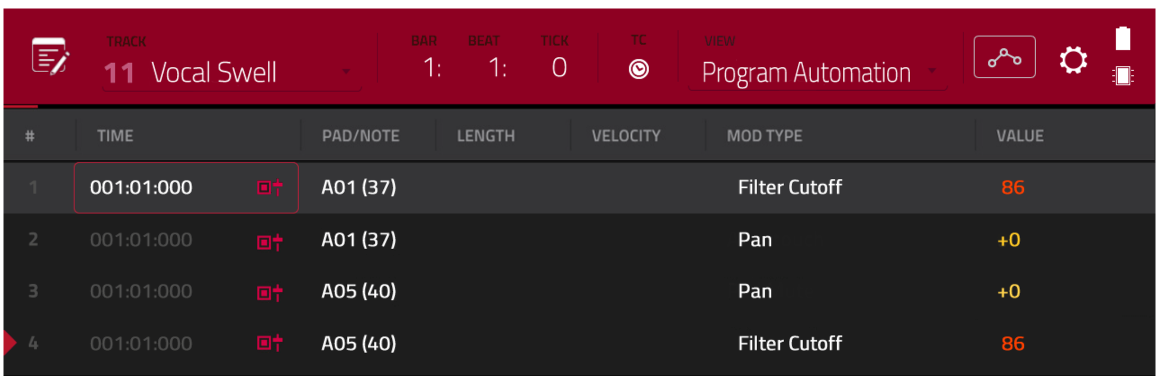
Now, the vocal sample is triggered in bar 14, so with **track 11** selected in **MAIN**, 'Locate' to **bar 13**. Change the automation control in the top right of the screen from '**Read'** to red '**Write'**.

Place your fingers on the two Q-LINK dials you just configured and hit **PLAY**. Remember automation does not require 'OVERDUB', it will record during normal playback as long as we've set 'WRITE' instead of READ.

During the count in turn the dials 'hard' to the left and as you hear the vocal come in, start to turn the Q-LINKS aggressively to the right to produce the pan and filter 'sweeps'. Hit **STOP** and set the automation control back to '**Read'**. Locate back a few bars and hit **PLAY** to hear the sweeps in action. Remember there's no way to edit these sweeps from the hardware interface, so if you are not happy with the result, simply re-record. Experiment with different sweeps – for example, try sweeping one parameter clockwise, the other anticlockwise.

You can hear my version on **sequence 8** (**RPG – Sweeps**).

Go to **LIST EDIT**, **track 11** ('**Vocal Swell**') and choose '**VIEW: Program Automation**':

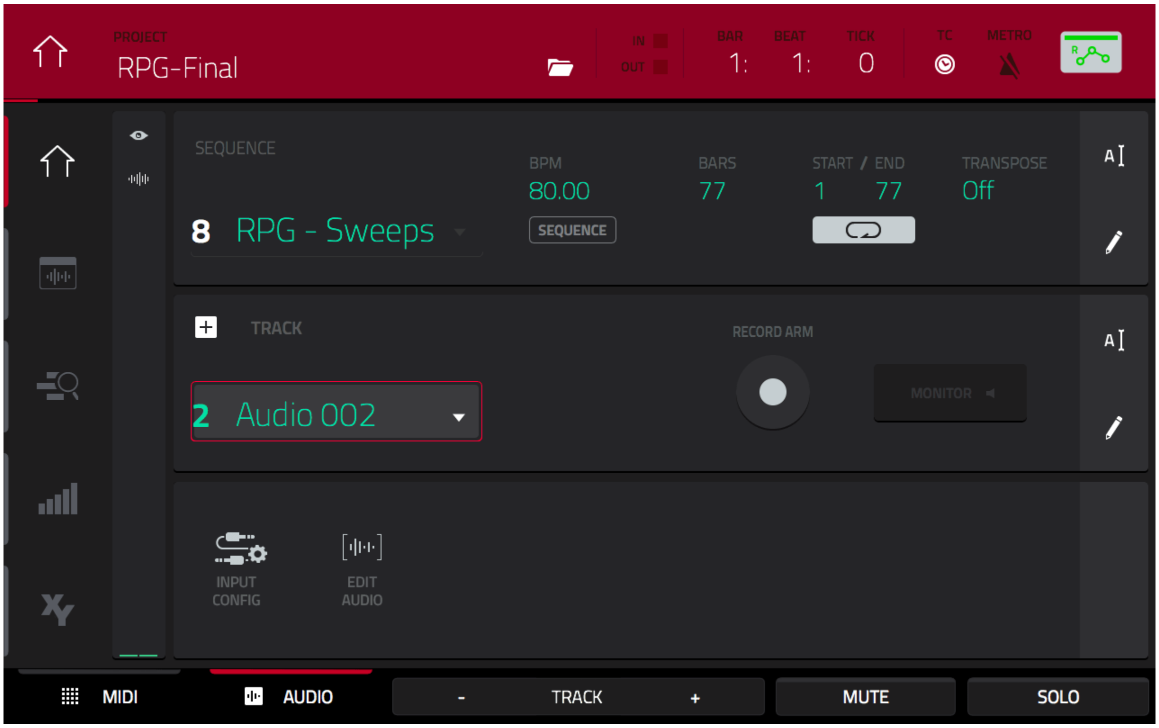


Here you can see your recorded '**FILTER CUTOFF**' and '**PAN**' automation events. You can also easily edit each event by tapping on the **PAD/NOTE** or **VALUE** parameters and adjusting using the standard methods (data wheel, +/- buttons etc). You can also use the **NUDGE** option to move the events as you would do with normal note events.

I find it very useful to tweak this type of 'Q-LINK' based automation data, especially pan events, as I find pan sweeps can be tricky to record without ending up with somewhat excessively extreme sweeps. So I just dive into LIST EDIT and reduce the VALUE for the more extreme pan events.

Adding Some Scratching

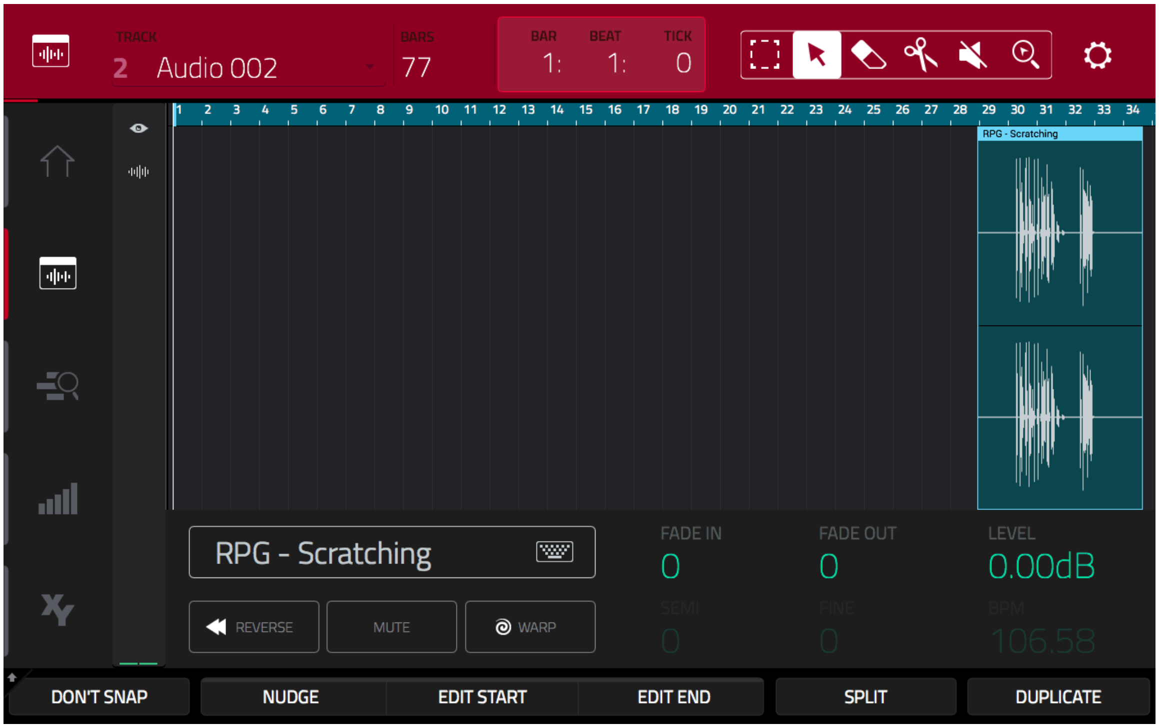
We can record some scratching to an audio track just like we did with the vocals. In **MAIN > AUDIO** hit the **+ icon** on the **TRACK** row to add a new blank audio track.



As we did with vocals, you can configure and arm this track to record a scratching performance, but let's just import some quick scratching I recorded earlier. Now in reality you'd probably add this scratching as a sample in a DRUM or CLIP program, but I want to use this audio to show a couple more features of audio tracks.

Hit the **EDIT AUDIO** icon down the left hand side of the screen and press **BROWSE SAMPLES**; select **RPG – Scratching**. Hit **DO IT**.

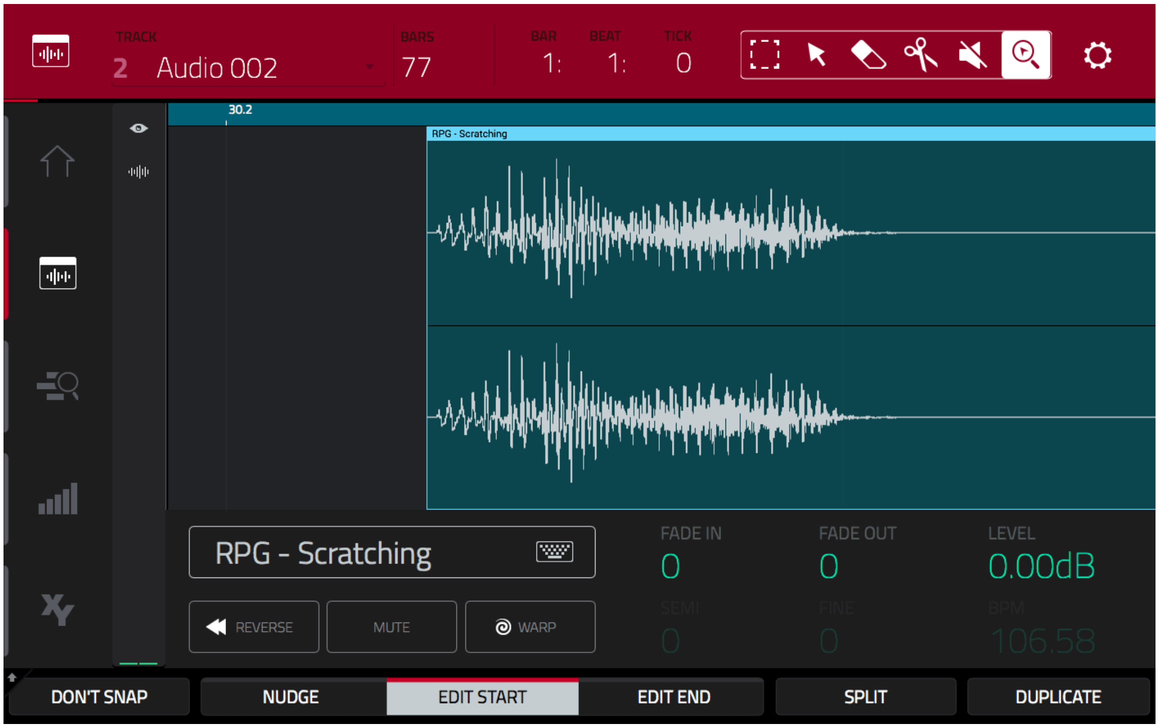
This has imported the scratches at the current playhead position (1: 1: 0). I want to add some scratching to the chorus which starts at **bar 29**, so we'll need to move this audio to that part of the sequence. Probably the quickest way to do this is to first select the zoom tool and 'pinch & zoom inwards' to reduce the magnification so the 29th bar is visible on screen. Now choose the '**SELECT**' tool from the top of the page and move the scratch audio to around the **29th bar**.



LOCATE to **bar 29** and hit **PLAY** to hear the scratching so far. It's unlikely to sound particularly great at the moment as we'll need to shift the audio until it fits nicely.

To make this easier, let's trim the start of the audio. With the audio still 'selected', hit the **EDIT START** button and begin turning the data wheel so the beginning of the audio file 'block' gets closer to the actual waveform. Use the **ZOOM** tool to get a closer look.

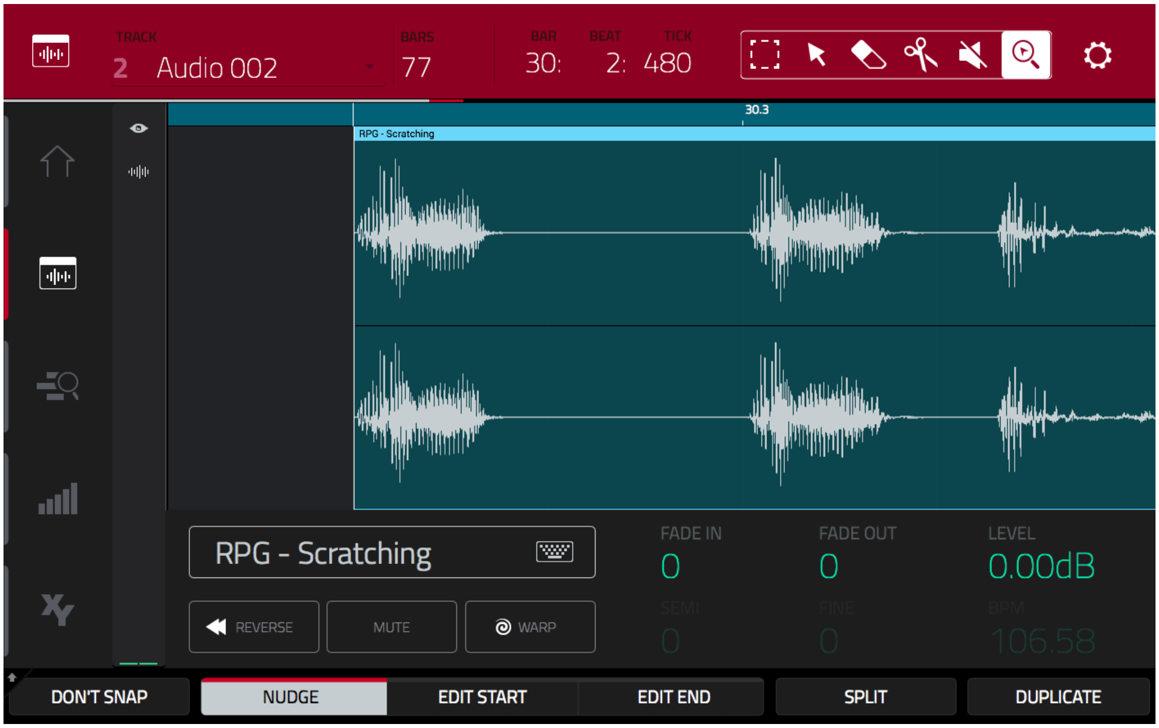
IMPORTANT: To get tight to the waveform, hold down **SHIFT** as you turn the data wheel.



The audio file is now edited nice and 'tight' to the actual start of the audio itself, so let's now move the start of the audio to the desired start position.

To move the audio region, either choose the **SELECT** tool and drag the audio (the higher the zoom magnification, the more accurate you can be), or alternatively, hit the **NUDGE** button at the bottom of the screen and use the data wheel to nudge the audio in T.C. steps.

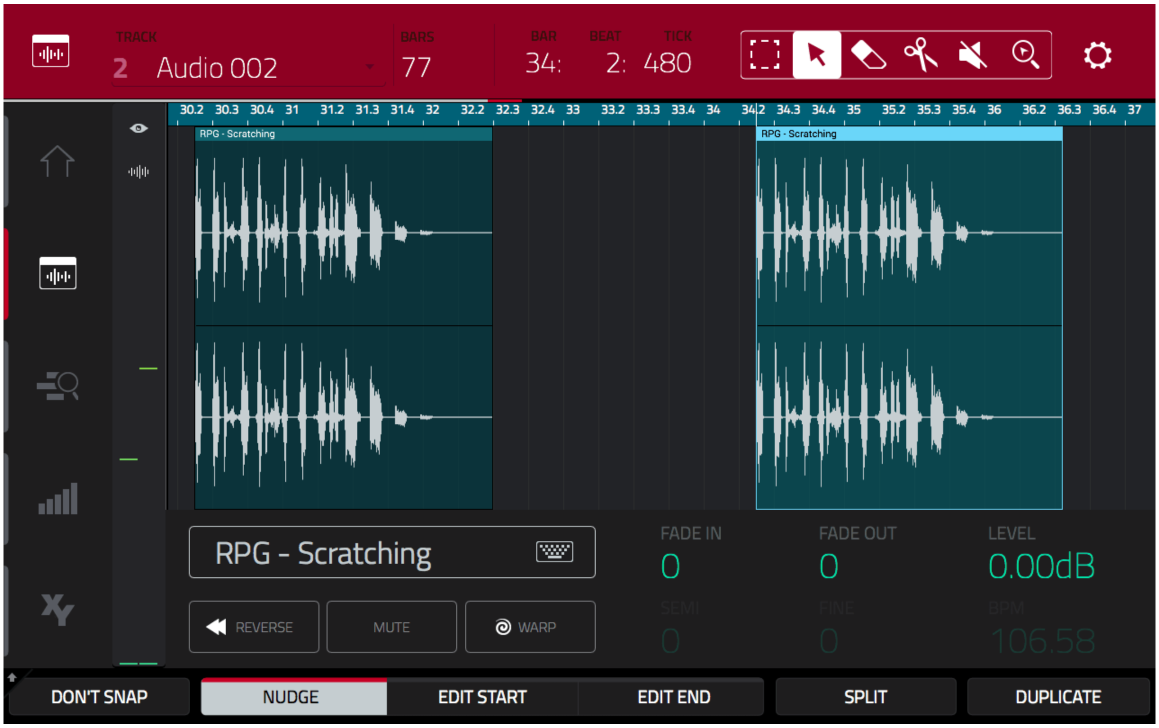
To move the audio to a specific time location I suggest first LOCATING the play head to the time location you wish to move the audio to. So LOCATE the play head at **30: 2: 480** and then use **NUDGE** or '**select+drag**' to move the audio to **30: 2: 480**.



LOCATE to **bar 29** and hit **PLAY** to preview the start of the chorus. I like the first part of the scratching, but not the 'Got it like that' scratch at the end. Tap on **EDIT END** and turn the data wheel anticlockwise so the audio snippet now ends before that last scratch:

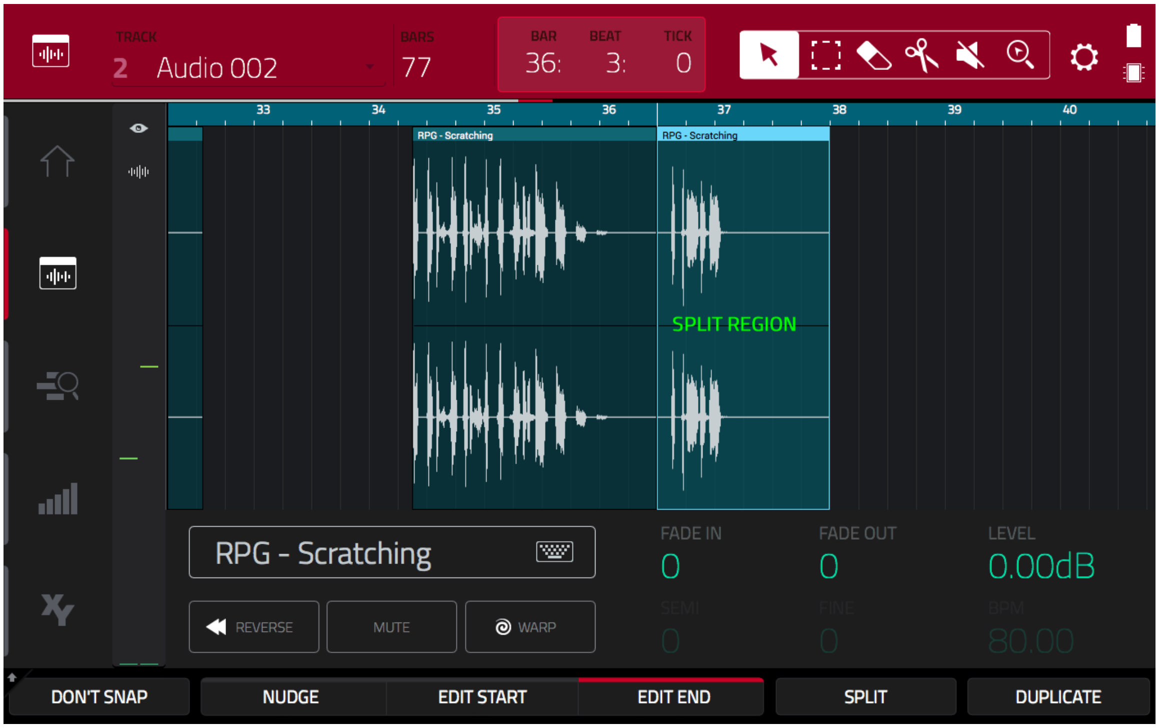
Let's now repeat this scratching later in the chorus. With your audio snippet still selected, hit the **DUPLICATE** button.

This has copied the selected portion of audio. We can now move this copied audio and place it in the second half of the chorus; move it to **34: 2: 480** - use **NUDGE** or use the **SELECT** tool and drag it.



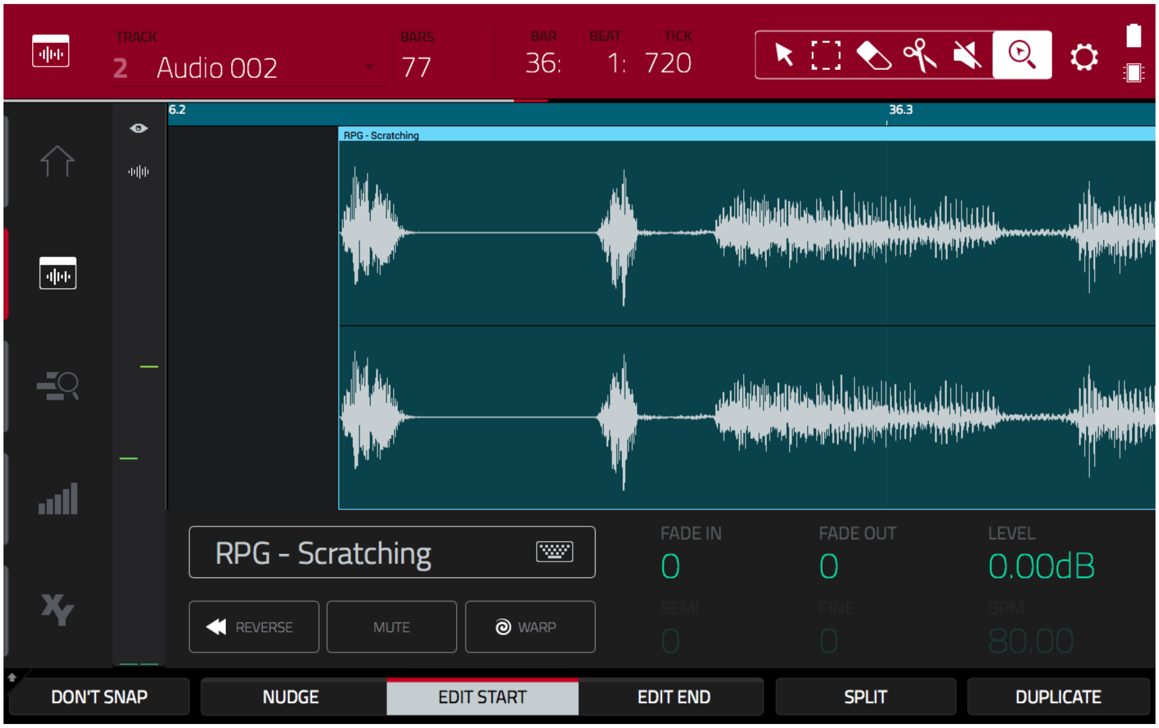
Remember that little scratch we removed with EDIT END? With your recently moved copy still selected, hit **EDIT END** once again and begin to **turn the data wheel clockwise**; you should see that little scratched phrase *reappear* – as you can see, any edits you make to an audio snippet's length are *non-destructive*.

Locate to the start of **bar 34** and press **PLAY** to hear the second lot of scratching. That last phrase isn't quite right as it's coming in too late, so with that second audio segment still selected, **LOCATE** to just before the small scratch phrase and press **SPLIT**; tap to select the newly created region:

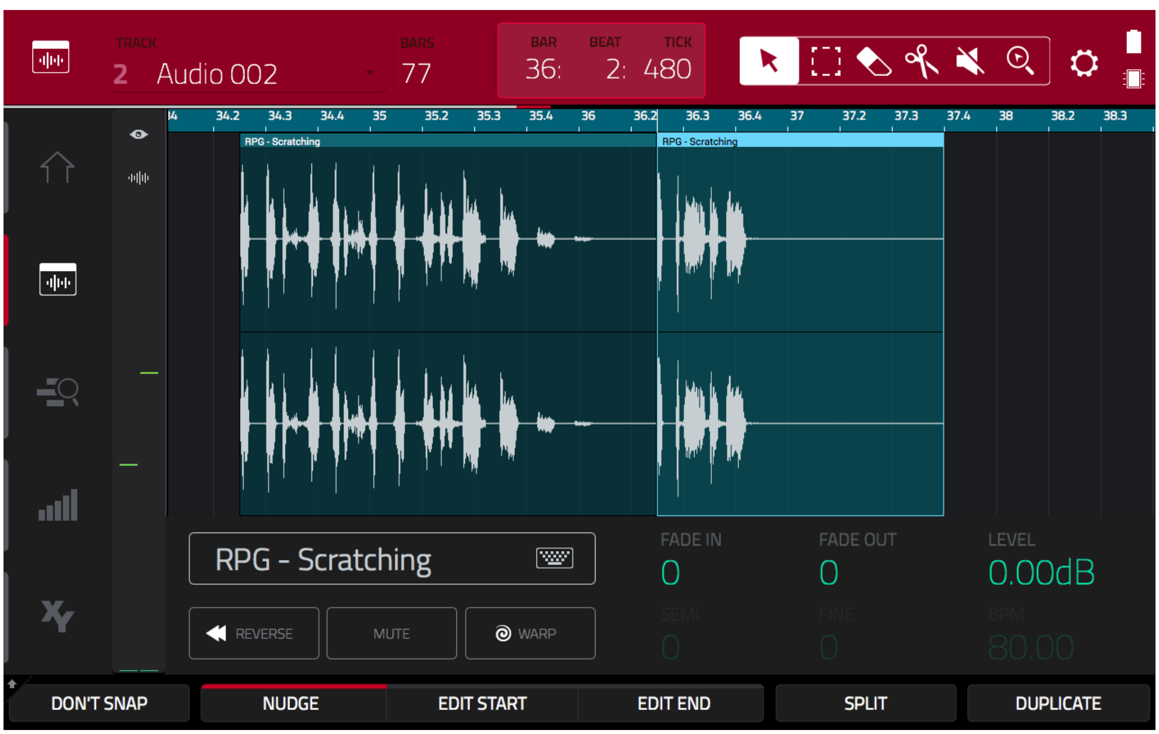


Alternatively, you could have selected the **CUT** tool (the scissors icon at the top of the page) and tapped where you wanted the waveform to be split - the **cut tool** is very useful when you wish to make a quick succession of cuts. For an accurate 'split' I prefer the SPLIT Tool.

Select the newly split off region, magnify it, hit **EDIT START** and trim the start of the region so the region begins where the audio starts:



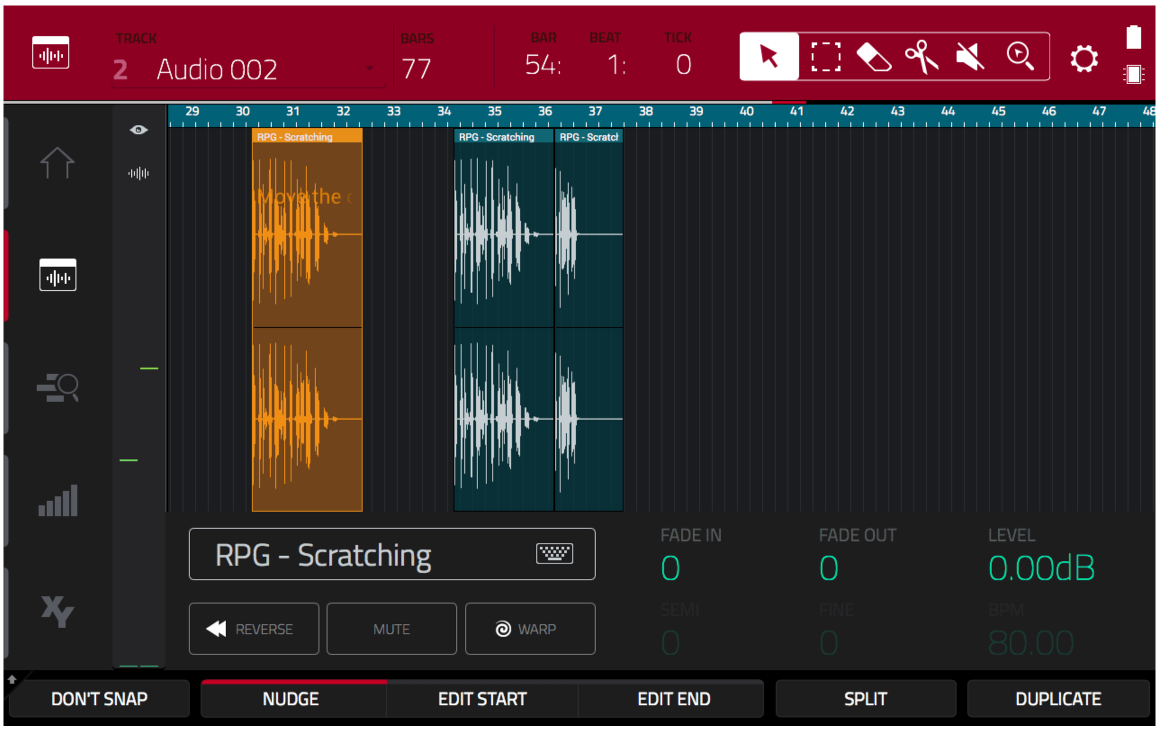
LOCATE to **36: 2: 480** (to set a position marker), hit **NUDGE** and move the split region back to **36: 2: 480**.



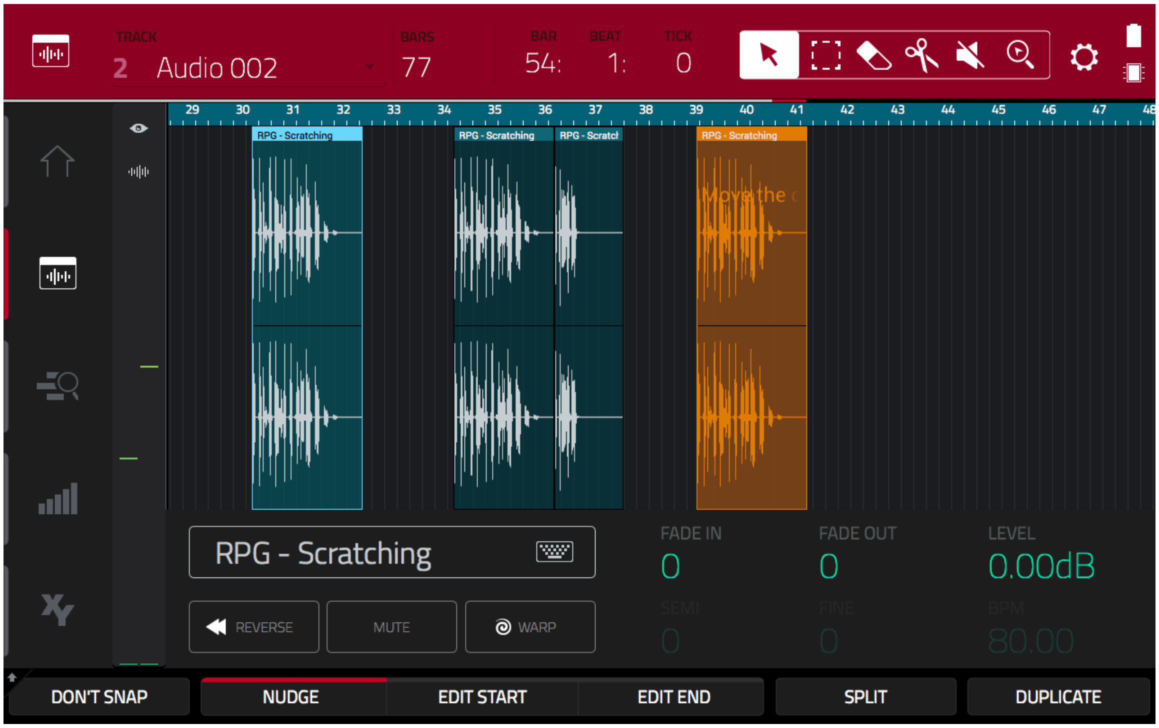
LOCATE to **bar 34** and press **PLAY** to preview the scratch. If you want to tweak the position of a cut, remember to hold down SHIFT when moving/nudging to move in single ticks.

Go back to around **bar 30;** let's copy this region to the second chorus (**54: 2: 480**).

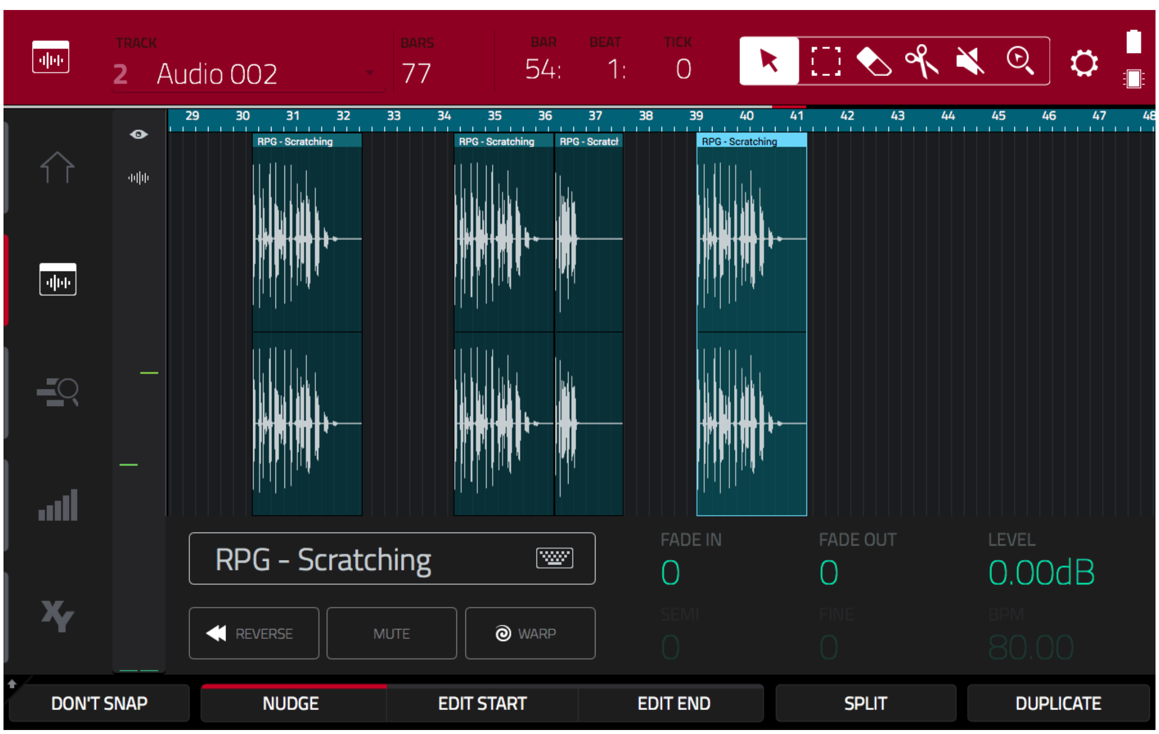
First, with the **SELECT** tool, tap in the snippet to select it. Now hit the hardware **COPY** button.



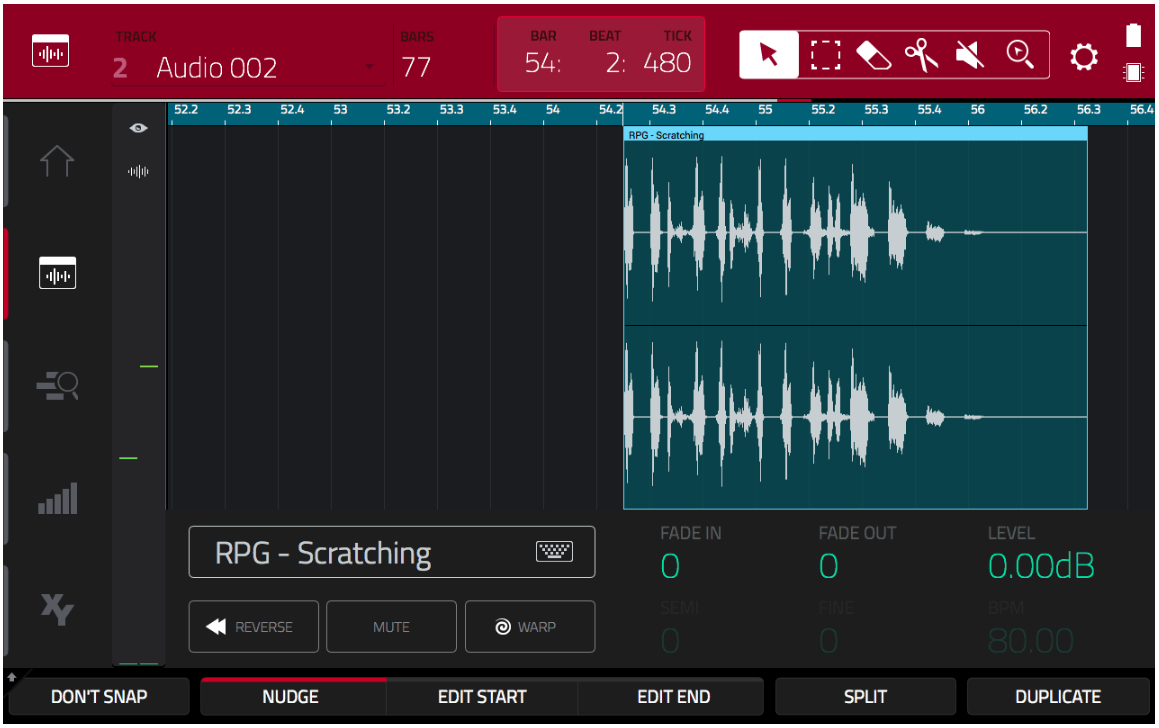
This has made a copy of the region to the clipboard. Now begin **turning your datawheel clockwise** - you'll see the orange 'copied' snippet begin to move along the timeline. To move it to **54: 2: 480** you can just keep on scrolling that data wheel until the copy reaches **54: 2: 480** or for a quicker workflow, just scroll until the copy gets to around **bar 39**:



At this point '**paste**' the copy to your timeline by pressing **down** on the **datawheel** so the wheel 'clicks'.



Now choose the **SELECT** tool and drag this pasted copy to **54: 2: 480**

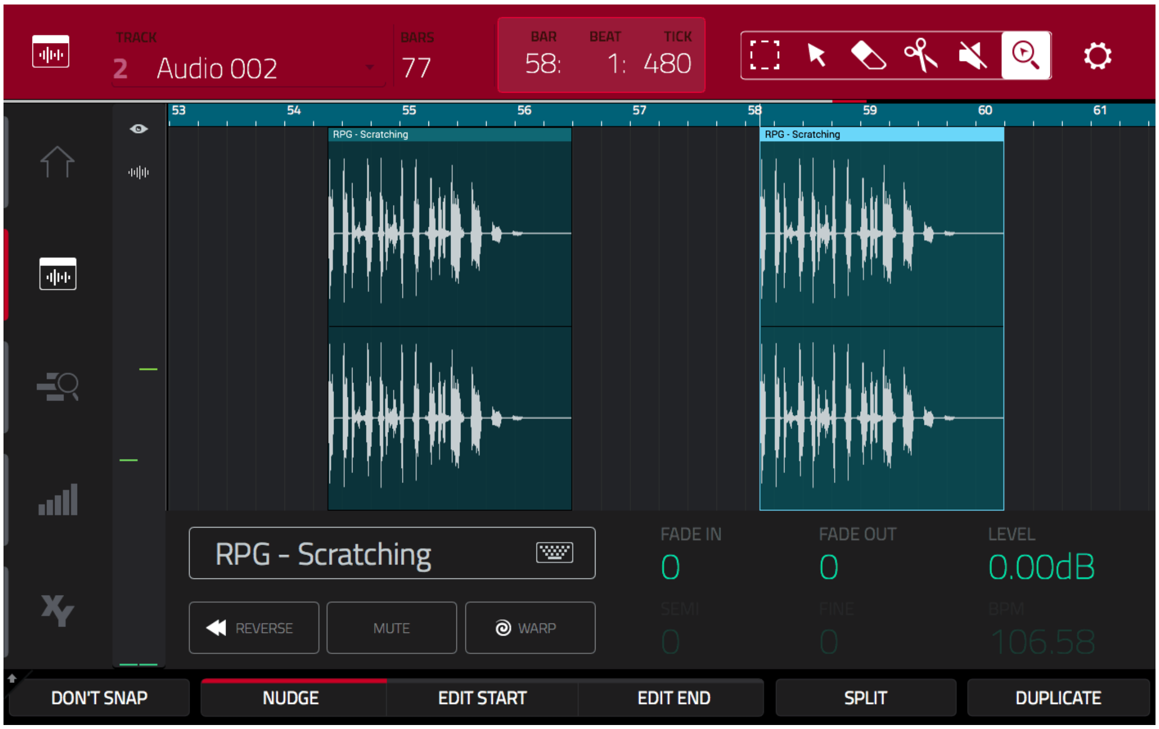


Using The Software COPY & PASTE Buttons

Now let's make another copy of this audio snippet and paste it to **58: 1: 480**. First make sure the snippet at 54:2:480 is selected and hold down **SHIFT**:

At the bottom of the screen you'll now see a new set of buttons appear; **CUT**, **COPY** and **PASTE**. Tap the **COPY** button to place a copy of the current region on our clipboard.

Now **LOCATE** to **58: 1: 480**; hold down **SHIFT** again and this time tap the **PASTE** button.

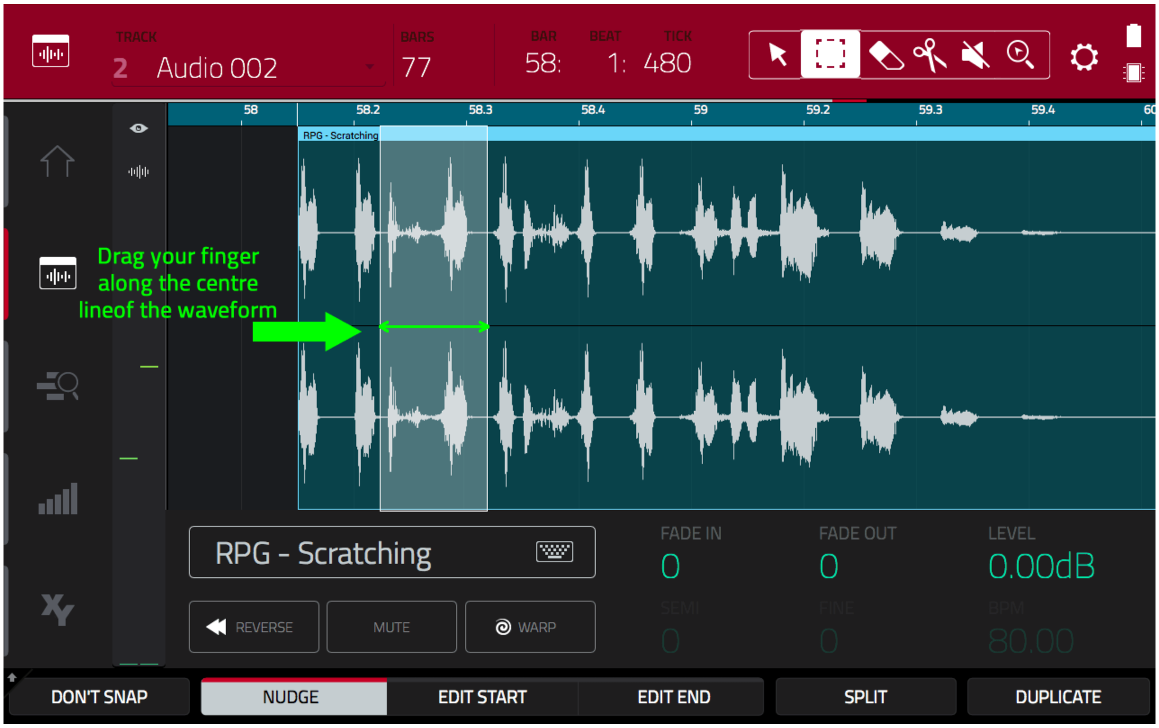


Using the Marquee Tool

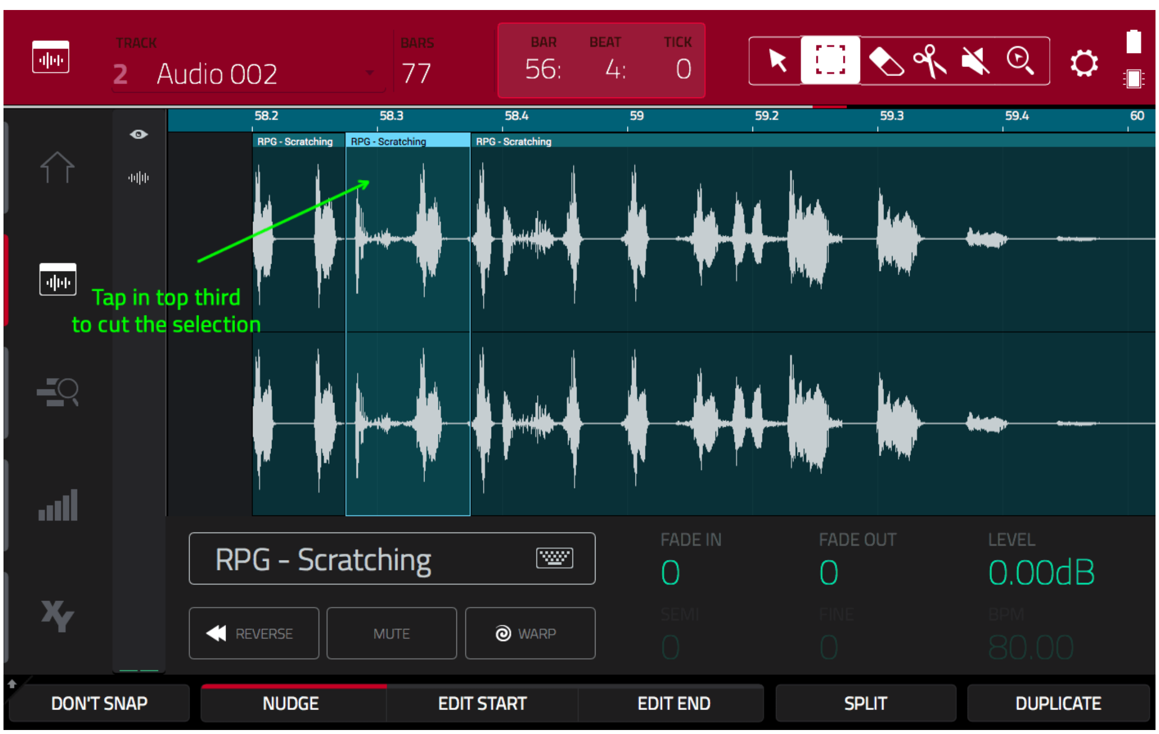
Let's copy and paste a small portion of our current region. To select the portion that will be copied, let's use the **MARQUEE** tool, which combines multiple tools together and therefore can speed up your editing workflow.

Firstly, magnify your waveform so the region at 58:1:480 is easy to edit accurately. Now choose the **marque tool** from the toolbar.

The first job we can perform with the marquee tool is to select a specific region from our waveform. To do this place your finger within the **centre line portion of the waveform** and drag across to create a selection within the waveform:

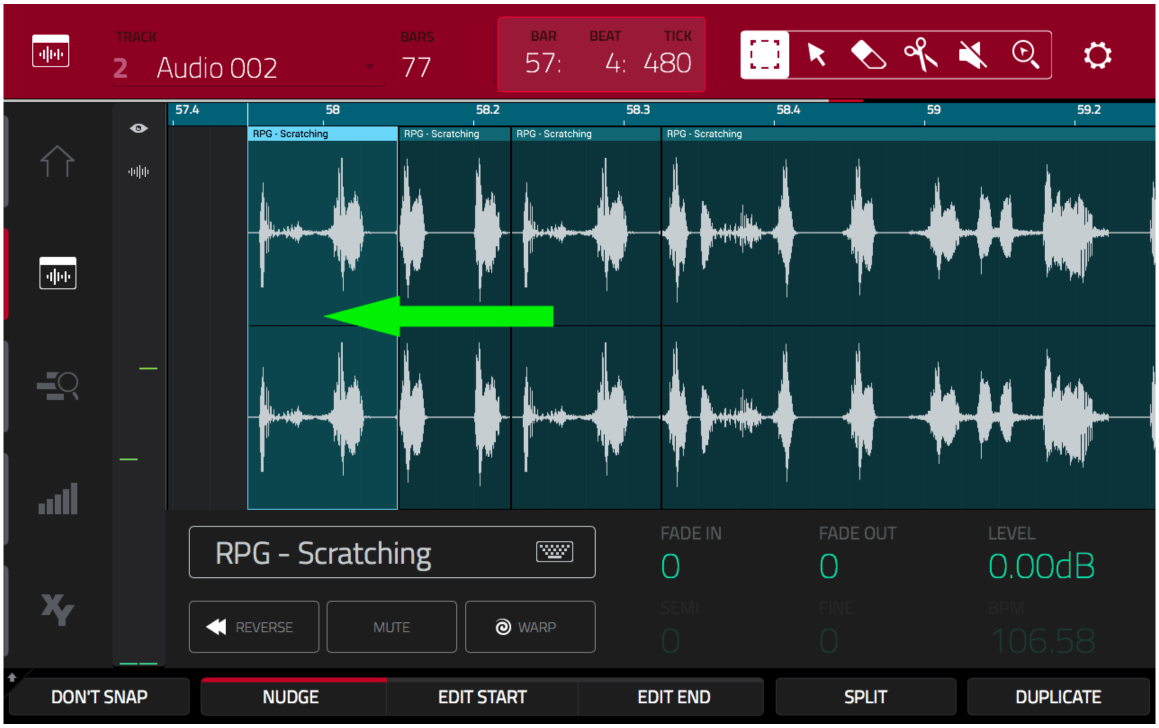


Hold down **SHIFT** for more accurate dragging. In the screenshot above I have selected a region representing the third and four peaks in the waveform. With the selection made, **tap in the top third** of the selection:



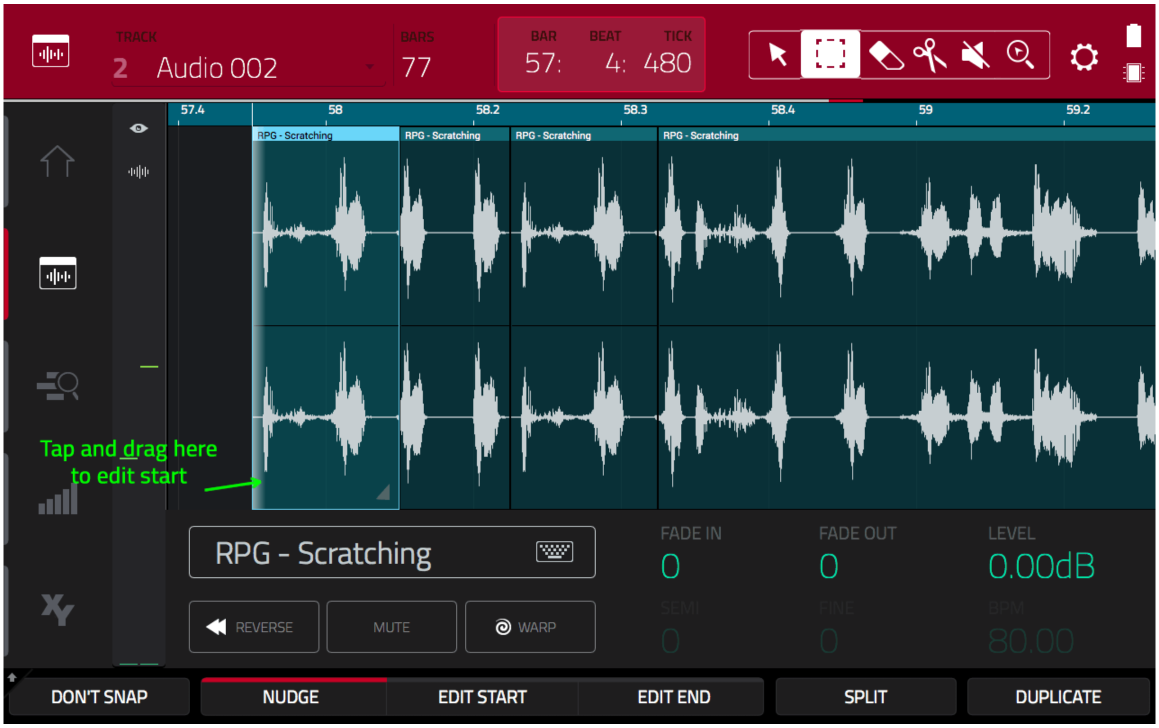
Tapping in the top third of the waveform (when using the marquee tool only) cuts your selected audio, creating a completely new region from that selection.

Hold down **SHIFT** and select **COPY**. Now **LOCATE** to **57: 4: 480**, hold down **SHIFT** and hit **PASTE**.

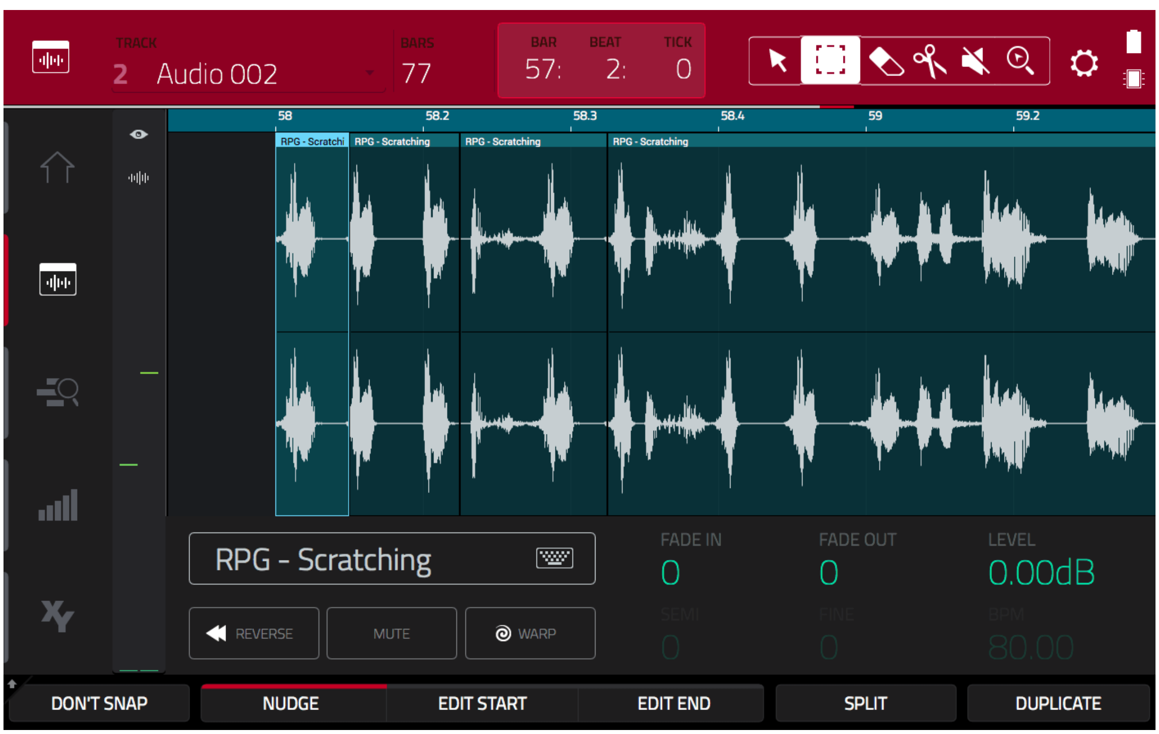


A third option when using the marquee tool is to edit the start and end of the region. This is achieved by dragging your region in the **bottom third of the waveform**.

To change the start of your region, place your finger at the beginning of the recently moved region, near the bottom; you should see an arrow appear and a faded edge will appear at the left side of the region:



Drag across to the right to change the start of this region - remember holding down **SHIFT** gives you maximum accuracy!



Locate to bar 57 and press **PLAY** to hear the new edit. With the marquee tool you can perform multiple edits without the need to keep changing tools, just remember the way the tool is 'split'; the middle third is for selecting your region, the top third is for cutting your region and the bottom third is for editing the start and end points of the region.

CUT vs COPY

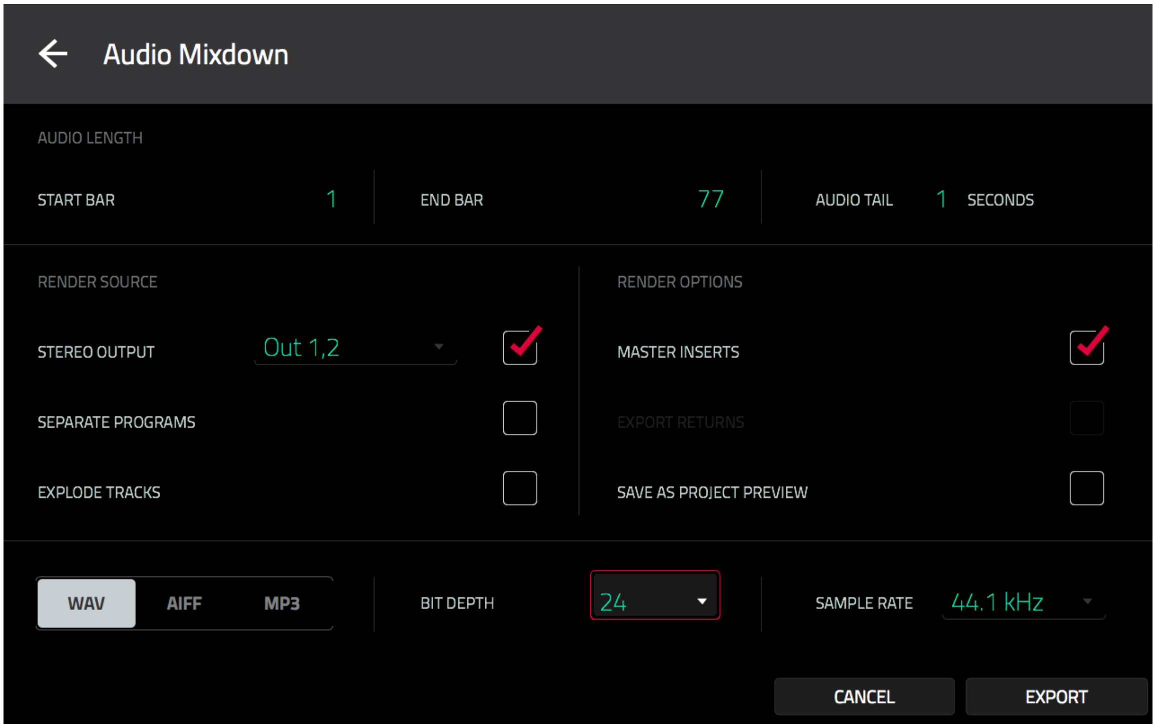
In addition to simply copying a region you can alternatively ‘cut’ it using the CUT button. This will still copy the region to your clipboard but it will simultaneously delete the existing region from the audio track. The resulting ‘cut’ can then be pasted anywhere you wish. In this respect, ‘CUT’ is another way of ‘moving’ a region.

Check out my version on **sequence 9**, **RPG – Scratches** (the scratches are on audio track 3). Here I’ve copied and pasted the various scratch regions throughout the rest of the track, tweaked the track LEVEL and added some delay using the delay already available via SEND 2.

Exporting Your Work

So, you’ve written a great track, mixed it down, tweaked it and polished it. Now what? How do we get your masterpiece out of the MPC?

In **MAIN**, hit the **PROJECT** folder at the top of the screen and hit the **EXPORT** button at the bottom right of the page to bring up the **Audio Mixdown** screen:



Here you can easily create a **stereo master** which is effectively a stereo 'bounce' of your entire song into an audio format of your choice.

By default, the **AUDIO LENGTH** is set to cover your entire song. The **AUDIO TAIL** is used to ensure that the export captures the sounds that continue to play after the sequencer has stopped; this could include reverb and delay tails and long 'one shot' samples for example. I tend to set this to at least 3 seconds.

Under **RENDER SOURCE** you choose exactly what type of export you are looking for. To create a stereo master, we just need the **STEREO OUTPUT** checked and for this song and as all our sounds have been left to the default output routing configuration, we can leave this set as **Out 1,2**.

Under **RENDER OPTIONS** you can see that you have the option to include your master inserts, ensuring that any insert effects added at the master stage are applied to the mix down.

Finally, select **WAV** format, a **24 BIT DEPTH** and whichever sample rate you prefer. This will produce a high quality stereo master for distribution – you can also select **mp3** format to create a reduced size version for upload to streaming web sites such as Soundcloud and Bandcamp.

To create your stereo master, simply hit **EXPORT** and choose a location for the WAV file to be saved to.

Getting Files 'Out' of Your MPC

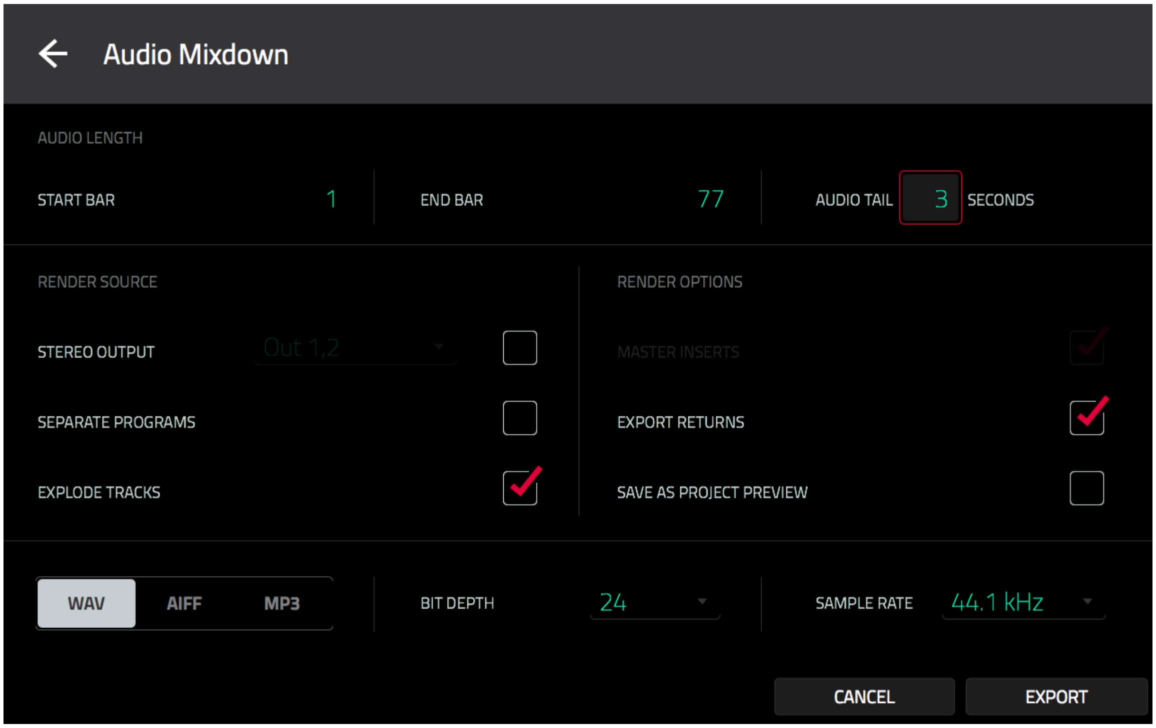
Currently it's not possible to directly access the internal 16GB drive in controller mode, so to make life easy, export to either a removable external disk or to the internal SATA (if you installed one). See **Appendix B** for more details on this.

Exporting Track Stems

While it's a fantastic sampling groove box, the MPC has some way to go before it can compete with some of the more mature, feature-packed Digital Audio Workstations (DAWs) such as Logic Pro and Pro Tools, so it's very common for producers to 'track' their MPC songs over to their favourite DAW to complete the entire composition in that highly optimised environment.

Transferring to a DAW requires we first convert each and every track in our song into separate audio tracks; these audio tracks are then imported into your DAW where you can continue the mixing process.

Return to the **EXPORT** screen.   This time instead of 'STEREO OUTPUT as the **RENDER SOURCE**, select '**EXPLODE TRACKS**.



This will create a rendered audio track for each unique track in your sequence, so after rendering you will have separate audio files for the kick track, snare track, hats, bass, piano, guitar, fx1, fx2, the reverse piano sound and the pad sound and so on. Now we can see the massive benefit of ensuring each track in our sequence contains only events for a particular instrument; once inside your DAW you're still going to retain maximum independent control over every element in your mix.

You can also optionally select to '**EXPORT RETURNS**' which will provide you with separate audio files for the effected returns data (which in the example in this chapter was our medium room reverb and a delay). Use this if your final DAW-based mixdown just doesn't sound right without the exact effects you used inside the MPC.

Project Previews

If you check the '**Save As Project Preview**' option the MPC will instead create an audio file that will be saved inside your current project folder for the sole purpose of creating an audio preview when you initially select your project file in the BROWSER (assuming you have 'AUDITION' enabled). This is worth doing for each and every project you create as it gives you an instant reminder of what the project sounds like before loading).

Fading Out a Song?

While we can 'perform' live parameter adjustments to all parameters in the MPC, you may have noticed that the ability to 'write' automation data is not possible in all screens.

Currently we can write automation events performed on program parameters including some insert effect settings), MIDI tracks and audio tracks. However we cannot for example, record any parameter changes made in the MASTER channel. This presents a problem, for example, if you wish to gradually 'fade out' your song at the end (which is a very common mixing requirement!).

While we wait for MASTER channel automation, the workarounds for fading a complete song are:

- **Fading each individual track separately** – as it's not currently possible to 'group' tracks together, this literally means selecting each track in the **CHANNEL MIXER** (all MIDI and all AUDIO tracks) and trying to record the same fade out on each track, one at a time!

- **Fading a stereo master** – perform a standard 'stereo output' export of your (un-faded song) as a 24 bit WAV file. Load the export into memory, and in a blank sequence of your project, load this exported file into an audio track. Audio track 'level' can be automated, so in **MAIN > AUDIO** enable automation writing and use **the bottom Q-LINK column 1** to adjust the track level to fade your track out. Now EXPORT this (single audio track) sequence to create the 'faded' master.

- **Export to third party software** – you can open your exported stereo master in an audio editor to draw a fade envelope at the end of the file, or export individual tracks to your favourite DAW, mix, master and perform the fade there.

026 Convert Plugins to Standalone Instruments

We've seen that we can't use plugin instruments while in 'standalone mode', so we previously 'bounced' plugin tracks to audio. However we do have another far more flexible option; we can quite literally '**clone**' those plugins by multisampling them and creating a native MPC 'keygroup' instruments from those samples.

The 'MPC2VST' Workflow

Using many of the techniques discussed in this book I've developed a 10 step workflow that handles the bulk of the 'cloning' work for you, it even renames all your samples and automatically maps them to a fully working keygroup program!

I'm going to run through a simple example first and then show you a second example that explains how the workflow can be easily customised to suit different types of sounds.

Before You Start

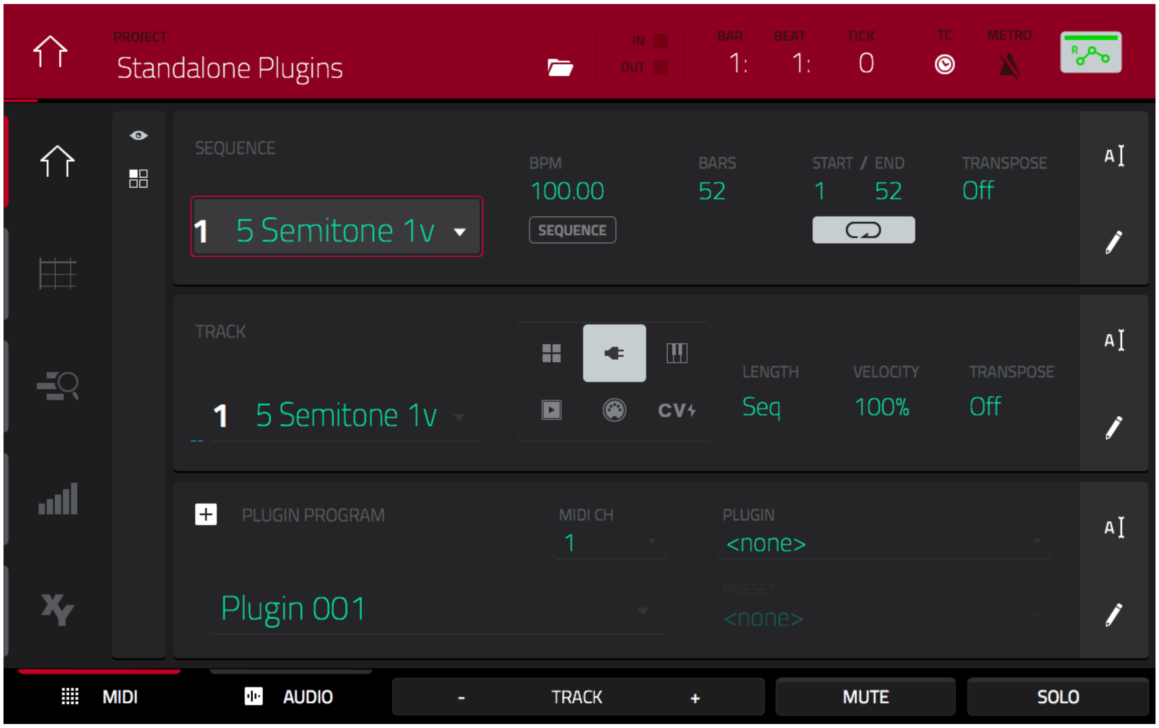
As we are initially going to be working with plugins you'll need to fire up **MPC Software 2.x** and connect your MPC via USB to your computer and turn on your MPC. Go to **MENU** and hit the **'MPC' chip** near the top right of the screen and select **CONTROLLER MODE**.

Now this workflow can be completed entirely in the computer GUI (there's one task that must be done in the GUI) but on the whole I'm going to explain the workflow from the hardware/touchscreen UI. You'll also need to copy the **Chapter 26** tutorial folder to your computer as it contains all the workflow files.

Step 1: Load the Project Template

Go to the **BROWSER** and navigate to the **chapter 26** folder. Enter the folder '**Standalone Instrument Plugins Workflow**' and load the '**Standalone Plugins.xpj**' project file.

The project loads up a 100 BPM sequence called **5 Semitone 1v** and sets **track 1** as a '**Plugin**' track.

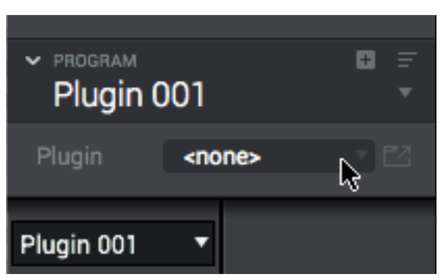


Step 2: Choose & Configure Your Plugin Patch

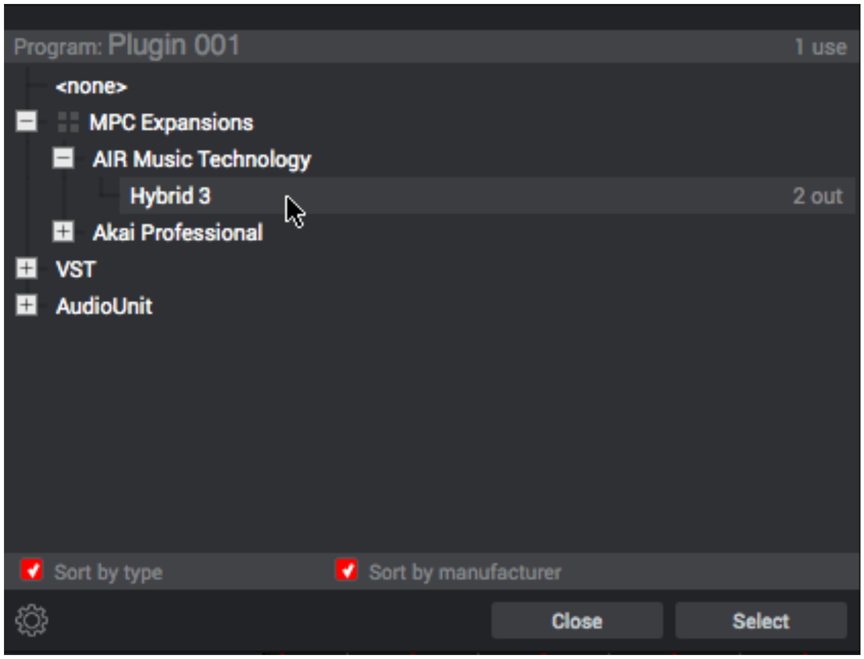
You'll now need to assign a plugin to this track. I'm going to use the '**Hybrid 3**' plugin simply because this is built in to all installations of the MPC Software, however once you're familiar with the workflow you can use this template to sample any patch from *any plugin* you wish; VST, AU and AAX - as long as the plugin is recognised by your MPC, you're good to go.

While you can configure your plugin from the hardware, this is the first part of the tutorial where I would recommend you head over to the computer GUI as it handles the plugin interface so much better than the touchscreen UI.

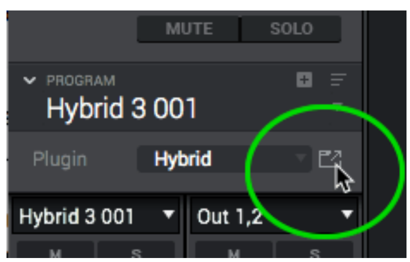
In **Main Mode** in the computer GUI, make sure the **Inspector** on the left hand side is open - if it isn't, hit **i** on your keyboard. Click the **PLUGIN <none>** box:



If you have **TYPE** and **MANUFACTURER** grouping enabled (bottom left of screen), you'll find Hybrid 3 inside '**MPC Expansions > AIR Music Technology**':



Double click **Hybrid 3** to assign it to our PLUGIN track. Now select a patch you wish to 'clone'. Click on the **Plugin Window** icon in the **Inspector:**



This will open the **Plugin Window.** Tap on the patch selector in the top right of the screen and from the '**Soft Leads**' group select '**13 Analog Square Lead**'.

Select pad **BANK D** or **E** on your hardware and play a few pads to preview the lead patch. Now when multisampling it's often a good idea to sample your instruments as dry as possible as you can always add effects like reverbs, delays and chorus to your MPC instrument program at a later date.

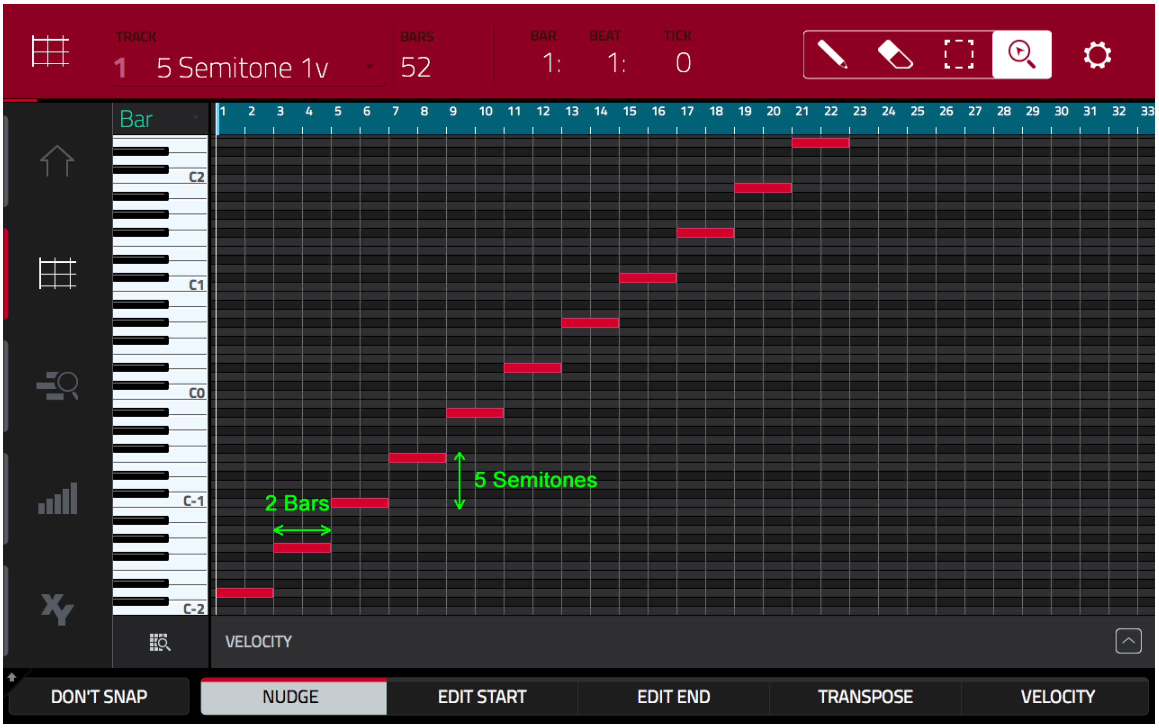
Head over to the **EFFECTS** tab in the Hybrid plugin GUI. At the top you'll see there are no effect inserts applied to parts A or B. However head down to the '**MASTER**' effects at the bottom where you can toggle through the three master effects using the tabs on the right; **CHORUS**, **DELAY** & **REVERB**.

As you can see, the **DELAY** and **CHORUS** are active, so for both effects click '**ON**' so the yellow light turns **off**:

Now preview the patch with your pads; the delay and chorus are now gone. Close your plugin window.

Step 3: Configuring Your Sequence

Back in the hardware UI, head over to **GRID VIEW** for sequence 1 (**5 Semitone 1v**):



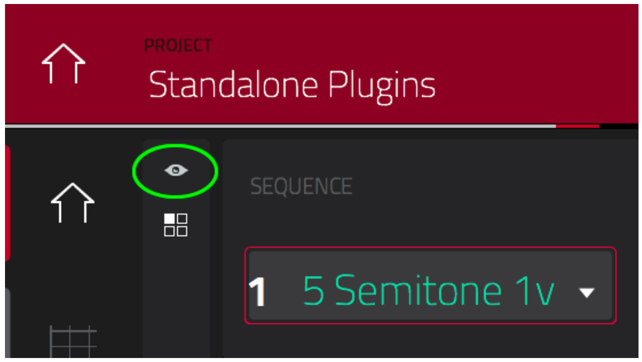
As you can see this is a 52 bar sequence containing **26** two bar long note events, each at full velocity. As the sequence name suggests, these events are added every 5 'semitones' and effectively cover the entire note range of the MPC. (The **1v** refers to this just recording a single velocity of each note).

Hit **PLAY START**; with the default tempo (100 BPM) each note in the sequence plays for 5 seconds. If you wish to adjust this you can just change the sequence tempo - pick a slower tempo of longer notes, faster tempo for shorter notes. I'll explain this in more detail in the second example.

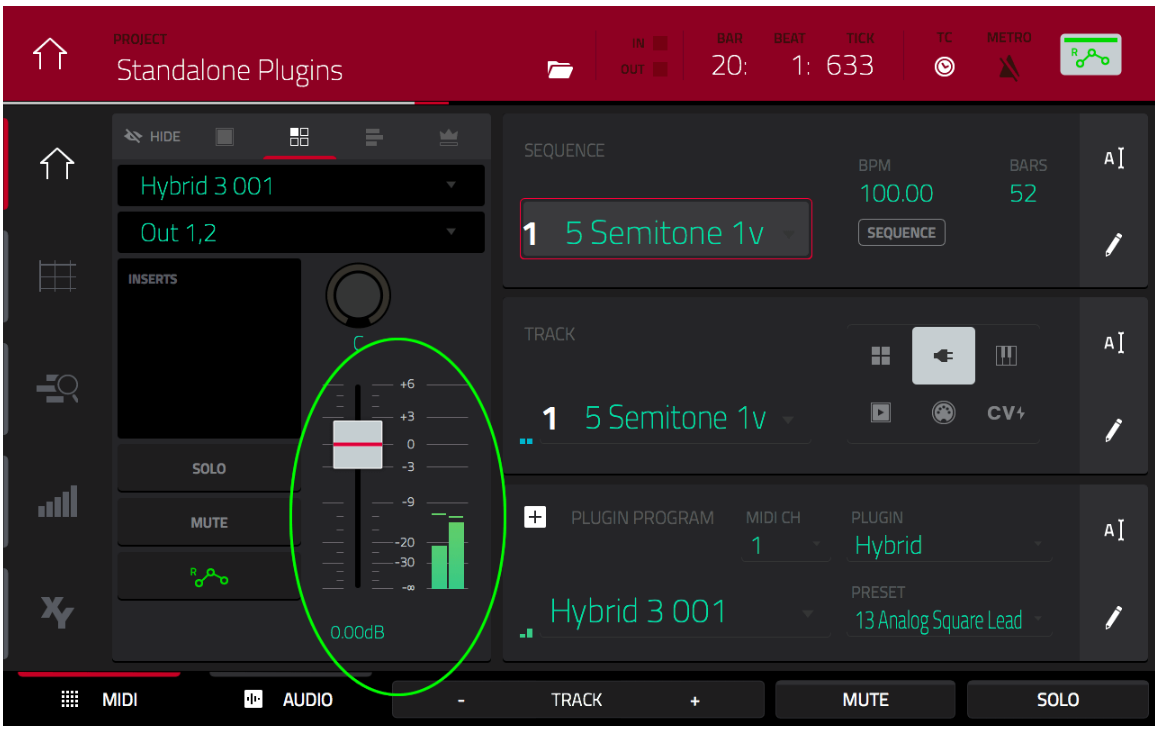
You've also probably notice that those really low bass notes in BANK A are just too low to be usable, but we'll easily deal with those later (the template system gets messed up if we start deleting bars from the beginning of our sequence).

Step 4: Check Your Levels

In **MAIN**, tap the 'eye' icon near the top left of the screen to reveal the program channel strip:



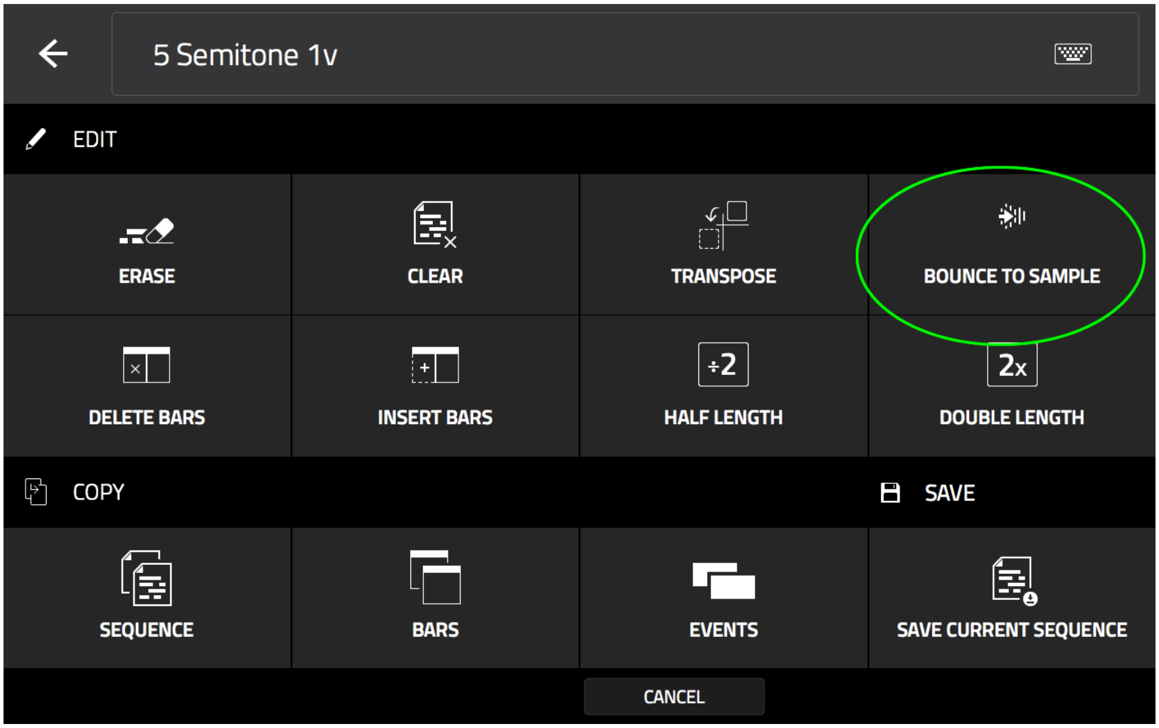
Hit **PLAY START** and observe your **LEVEL METER:**



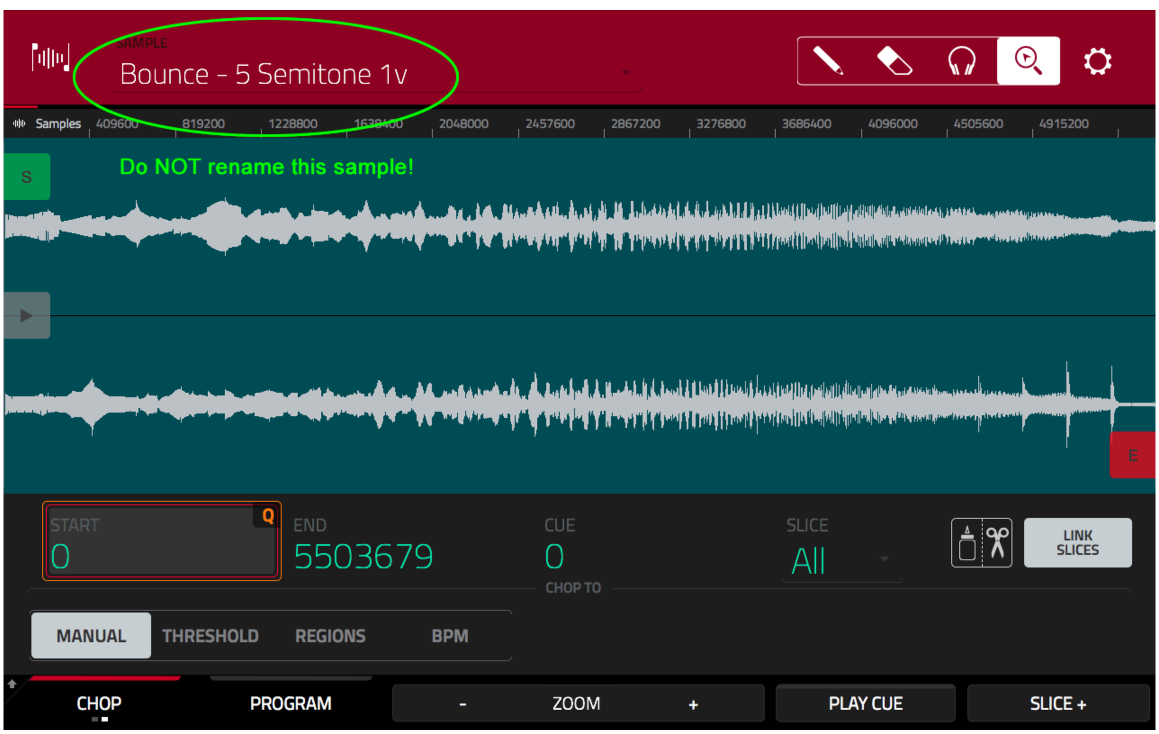
Adjust the slider to set a level that ensures there's no clipping (i.e. the levels don't go to the red). Alternatively, perform this in **MENU > CHANNEL MIXER**.

Step 5: Bounce Your Track

We're now going to bounce this sequence to create an audio sample from it. To do this in the hardware UI simply go to **MAIN** > **top pencil icon** > **EDIT** > **BOUNCE TO SAMPLE**



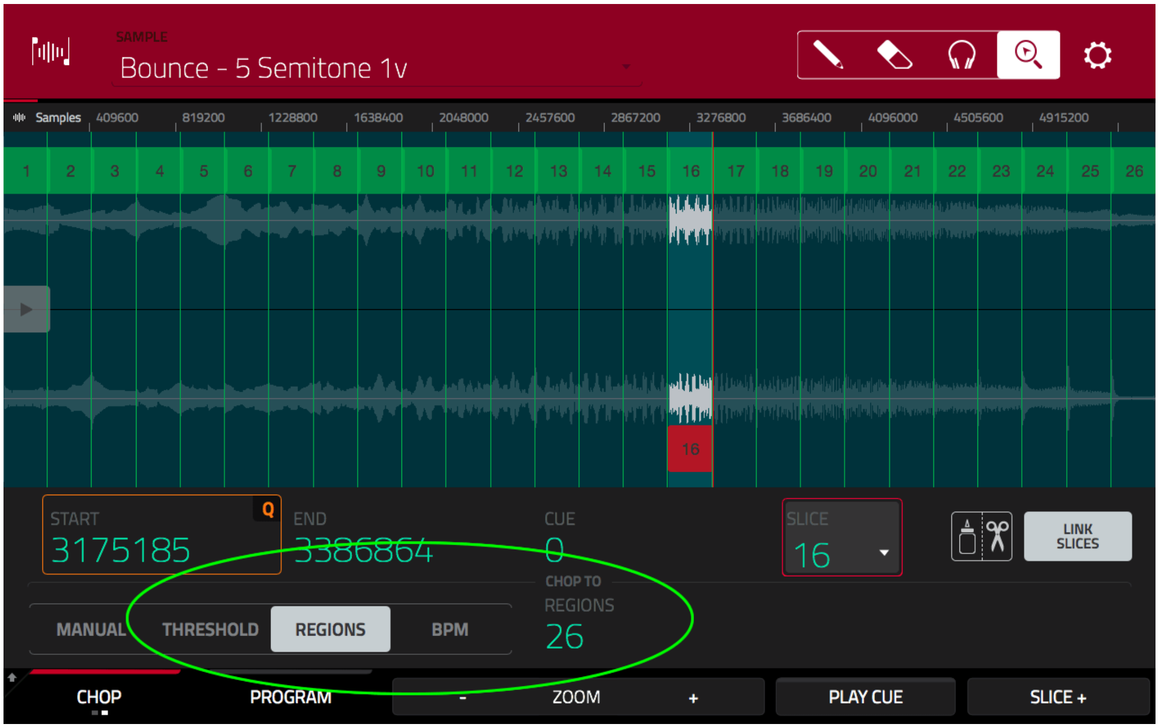
Head over to **SAMPLE EDIT;** there should only be one sample in memory '**Bounce - 5 Semitone 1v**':



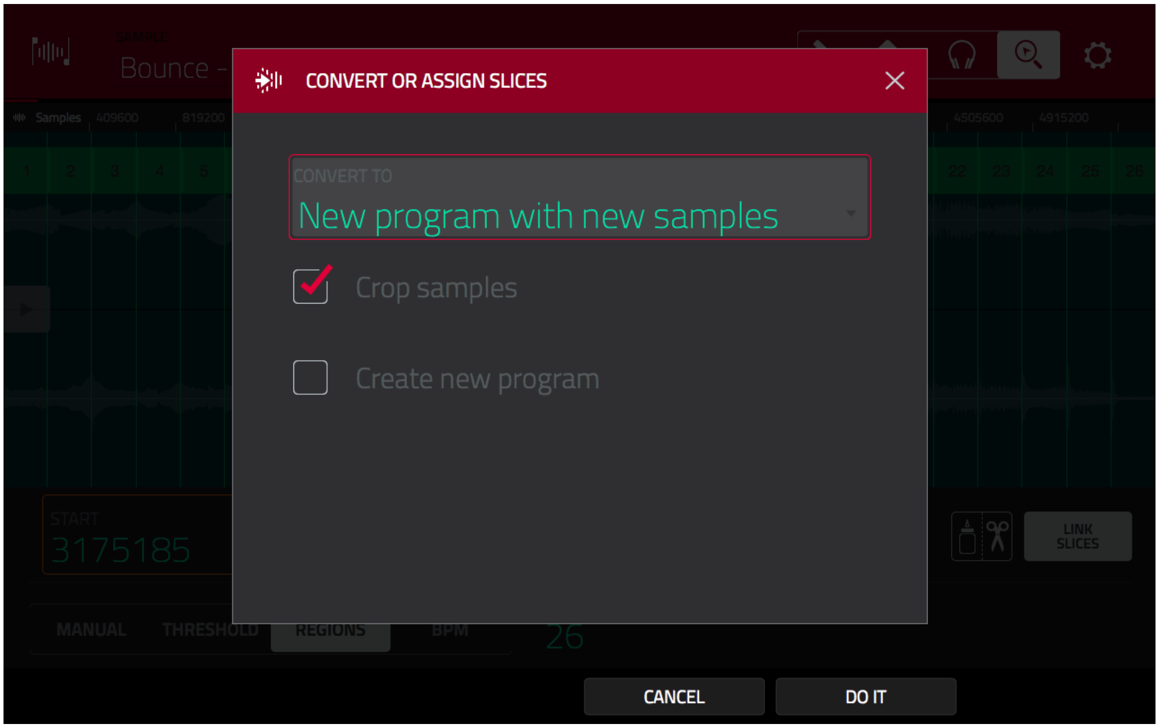
Important: Do **not** change the name of this sample!

Step 6: Create Your Chops

Press **TRIM** again to enter **CHOP** mode. Remember how our sequence contained **26 events**? The means we need to create 26 chop regions. Select **REGIONS** as the chop method and set **CHOP TO REGIONS** to **26**.

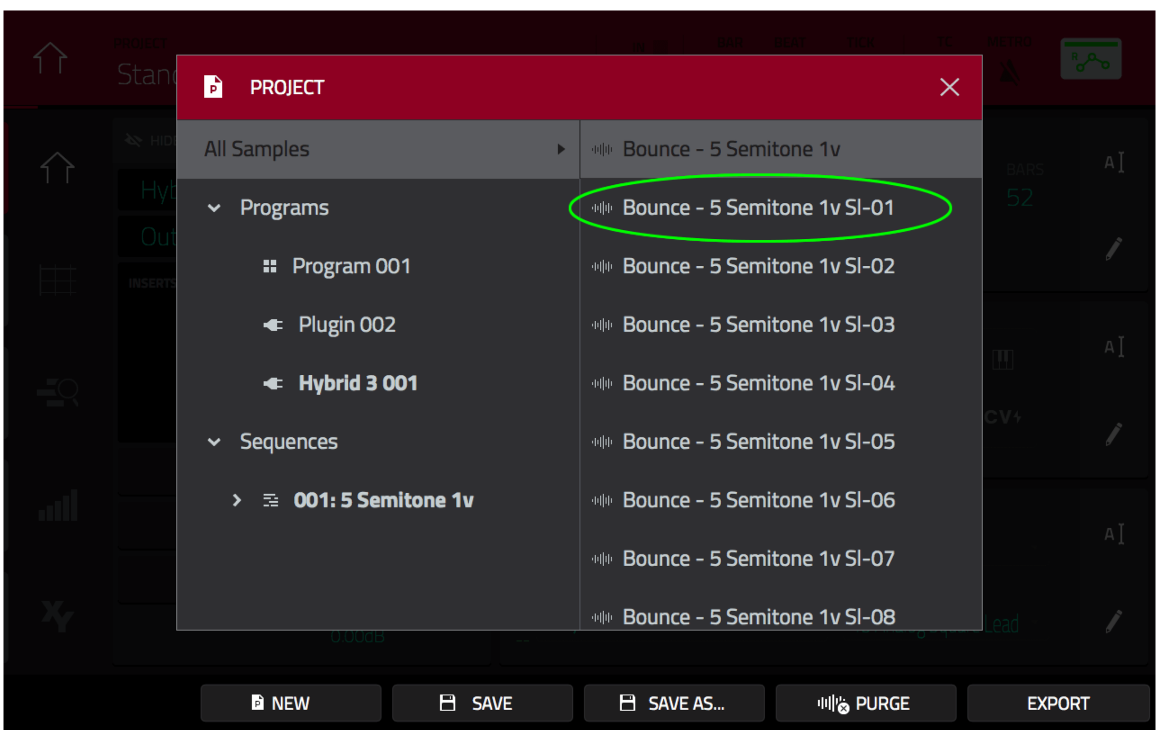


Perfect. Now hold down **SHIFT** and hit **CONVERT**. Choose '**New program with new samples**':



Make sure you **check** '**Crop Samples**' (we definitely want to create individual samples) and **uncheck** '**Create new Program**'. Hit **DO IT** to create your chops.

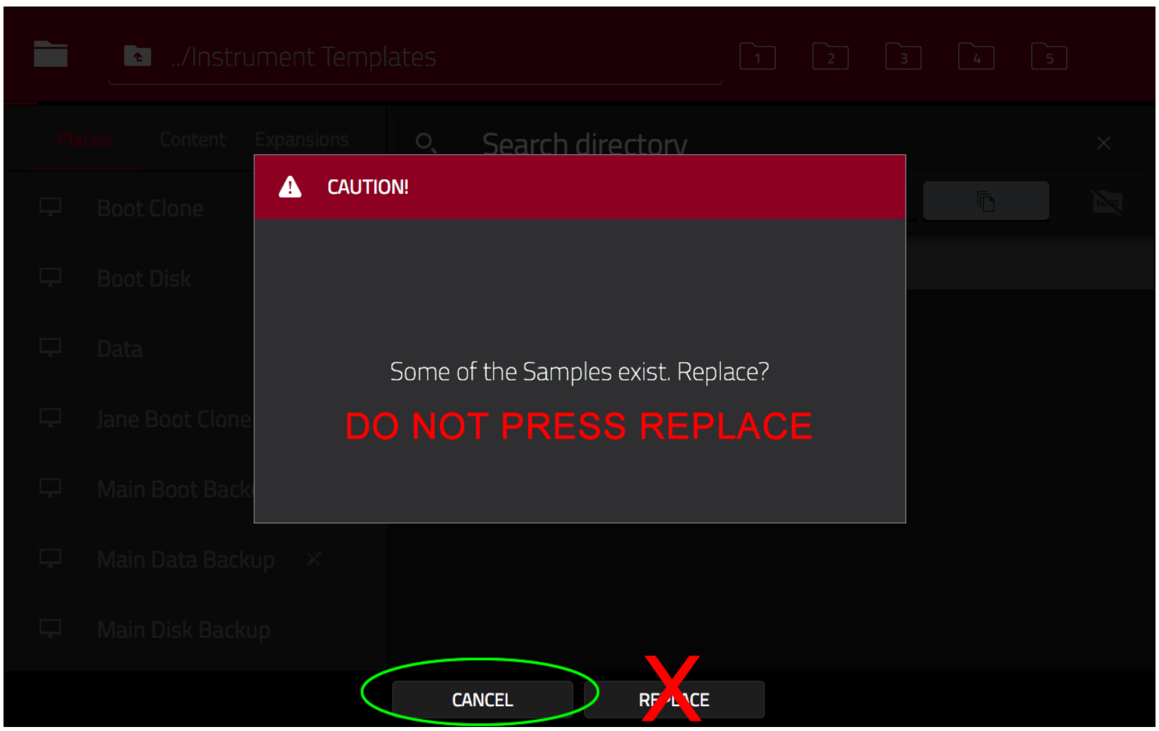
Go to **MAIN** and hit the folder icon at the top centre of the screen to view the **PROJECT** panel:



Make sure '**All Samples**' is selected on the left and you should see all your chops on the right. Each one of these is a note from your multisampled Hybrid patch.

Step 7: Import Your Instrument Program Template

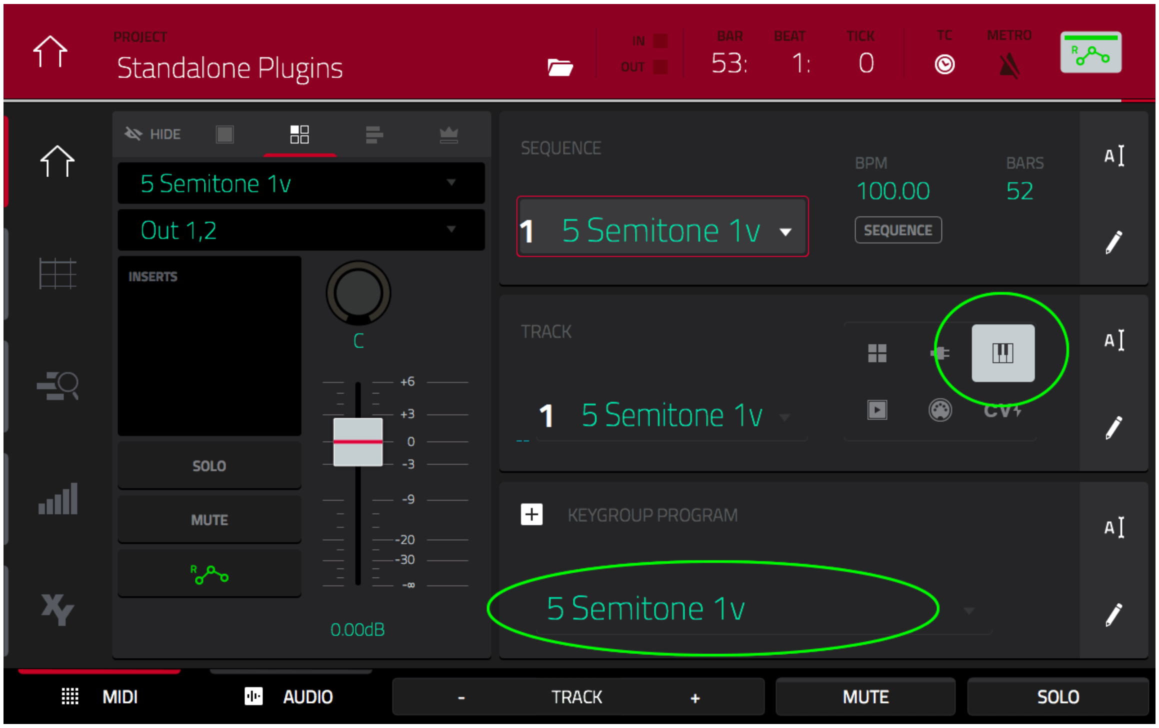
In the **BROWSER**, navigate to the **Chapter 26** folder and open the '**Blank Instrument Programs**' folder. Load the **5 Semitone 1v.xpm** program. Upon loading you will see this:



VERY IMPORTANT: It is vital that you select **CANCEL** here, do NOT press REPLACE. For some reason, on some systems pressing REPLACE causes the MPC to completely crash.

So press **CANCEL** - this will load the keygroup program into memory.

Now go to **MAIN**, change **track 1** to a **KEYGROUP** type track. The newly loaded **5 Semitone 1v** program should automatically assign itself to the track:



Here's the magic - select **BANK D** and start playing your pads; you should hear your new multisampled lead patch program in action!

How does this work? Well this '5 semitone' program template contains 26 key groups to match the 26 events in the '5 Semitone' sequence and each keygroup has a 'key range' of 5 semitones.

Now, remember I said that you must never rename the original 'bounced' sample? That's because when the MPC creates all those chops from that sample it uses a *predictable naming structure*, so we always know what the names of the 26 chops will be.

So in the program **keygroup 1** always looks for a sample **5 Semitone 1v Sl-1**, while **keygroup 2** looks for **5 Semitone 1v Sl-2** and so on.

When you load that program into your project the MPC realises that all the samples are already there in memory and immediately attaches them to your program.

This means you don't have to spend any time creating keygroups, assigning samples, or setting up root notes; the program does everything for you - but it only works if you don't change the name of the bounced sample (nor any other names in the templates!).

Dealing With Root Notes

As we discovered in chapter 10, keygroup programs know how to automatically tune samples to the correct pitch thanks to the 'root note' we set within the sample itself. However at no point in this workflow do we assign the root note to our sample chops.

This is because rather than set the root note at 'sample level' (as we did in chapter 10), this workflow assigns root notes at the *program level*. Take a look at at **keygroup 10** in **PROGRAM EDIT > PAN VELOCITY**:

Here you can see that in the **ROOT NOTE** column, **LAYER 1** has been set to a ROOT NOTE of **B1**. This means that any sample assigned to layer 1 of this keygroup will be assigned to be in the pitch of B1. By assigning root note this way, there is no need to assign the root note within the sample itself, which is perfect for this workflow.

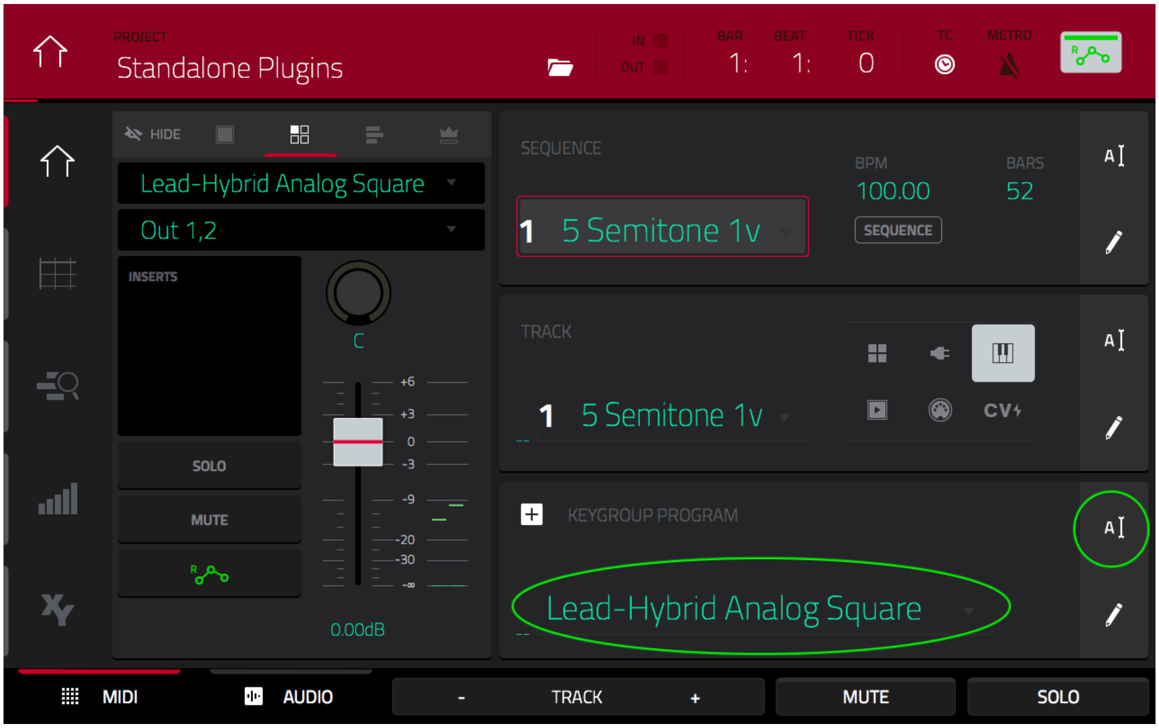
The clear advantage of working with 'program level' root notes is that the program itself handles all root note assignments and is therefore perfect as an instrument program 'template'.

Step 8: Tidy Sample Names

At this point your program and all the samples have the generic template names but it's ridiculously easy to fix this.

First, give your new keygroup program a more useful name, for example, 'Hybrid Analog Square', or if you want to use Akai's tagging standard, '**Lead-Hybrid Analog Square**'.

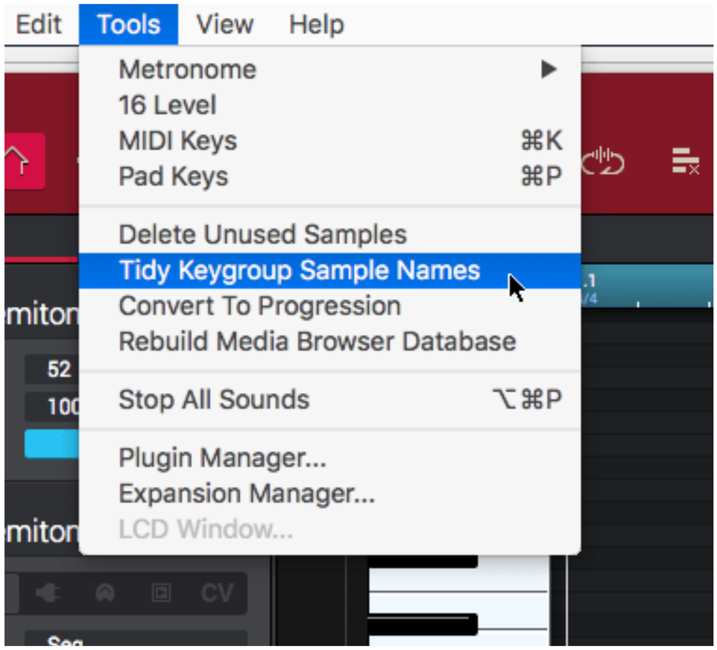
This can be done via **MAIN > KEYGROUP PROGRAM** row > '**A**' icon':



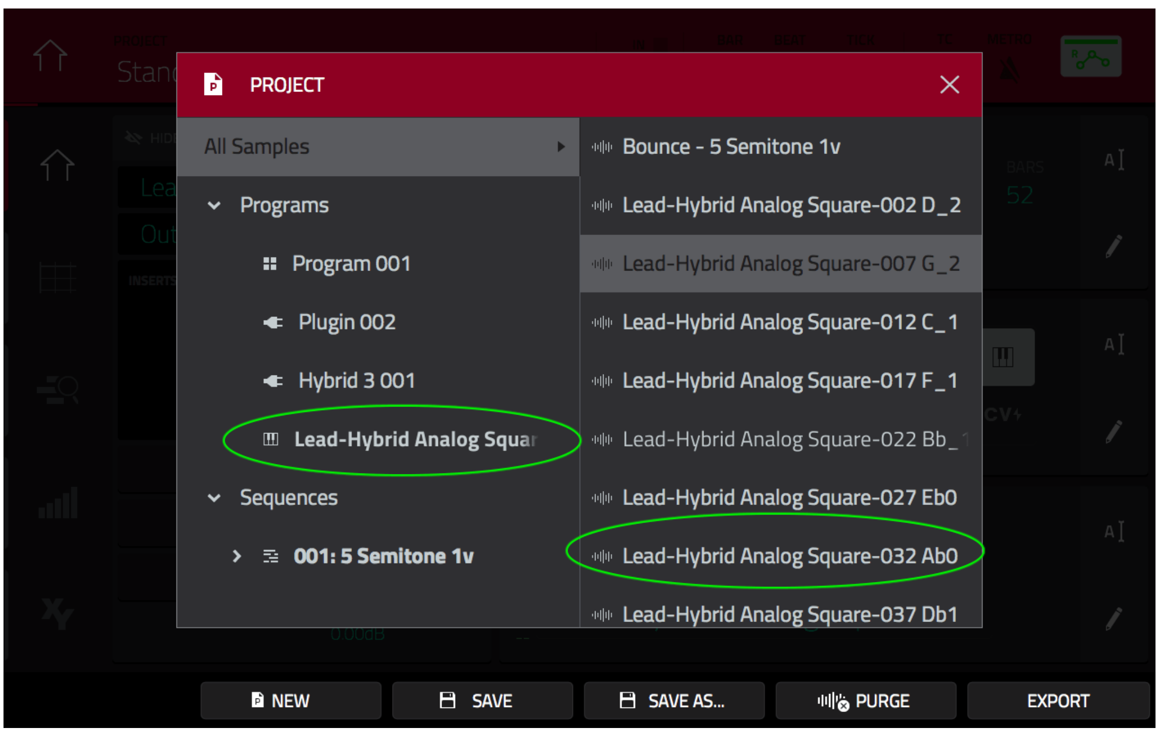
What is 'Tagging'?

Tagging is Akai's a way of sorting your sounds, programs and sequences into searchable 'groups'. This is primarily used in the MPC Software GUI within the Media Browser and the Expansion Browser, although tags can be useful for keyword searching within the standard browser. I look at pages in Expansion tutorial (Appendix C)

Now the clever bit. In the computer GUI, go to '**Tools > Tidy Keygroup Sample Names**':



Now take a look at the **Project** panel:



The MPC has now renamed all your note samples using the 'program name' followed by the the MIDI note number and pitch of the note the sample represents.

Step 9. Save Your Program

Don't forget to save this new program along with all its samples! You can right click on the program in the Project panel and select '**Save**', or in the hardware UI go to **MAIN**, hit the **pencil icon** on the **KEYGROUP PROGRAM row** and select **SAVE CURRENT PROGRAM**.

As part of the chapter files I've included a separate '**Saved Instruments**' folder which you could use to save instruments to, otherwise just save them wherever you wish.

Step 10: Reset The Project Template

Now that your plugin patch has been cloned and the MPC instrument saved, you can move onto multisampling another patch. But before you start, you'll need to delete the original 'Bounce - 5 Semitone 1v' file, otherwise the next bounce you perform will be given a name with a '1' appended to the end, i.e. '**Bounce - 5 Semitone 1v (1)**' .

I find the easiest way to do this is to just re-load the original **Standalone Plugins.xpj** file, either from the **BROWSER** or just drag it from your computer directly into the MPC Software GUI. Alternatively you could manually delete the original **Bounce - 5 Semitone 1v** file before starting a new session, but I find just reloading the original 'Standalone Plugins' project file is a bit quicker.

All done - you're now ready to clone your next plugin patch by repeating steps 1-10 as many times are you need to.

Example 2: Cloning a Bass Patch

Let's how easy it is to customise this workflow to suit ay type of sound. Make sure you've performed a 'reset' by loading the **Standalone Plugins.xpj**.

Assign an instance of Hybrid to track 1, this time select '**Basic Basses > 13 Knocking PWM Bass'**. As before, head over to the plugin's '**EFFECTS**' section and disable the **CHORUS**, **DELAY** and **REVERB**.

Press **PLAY START** and have a listen to the bass patch. Unlike the previous example, the notes in this patch do not play for as long as you hold down the pads, instead they fade out quite quickly.

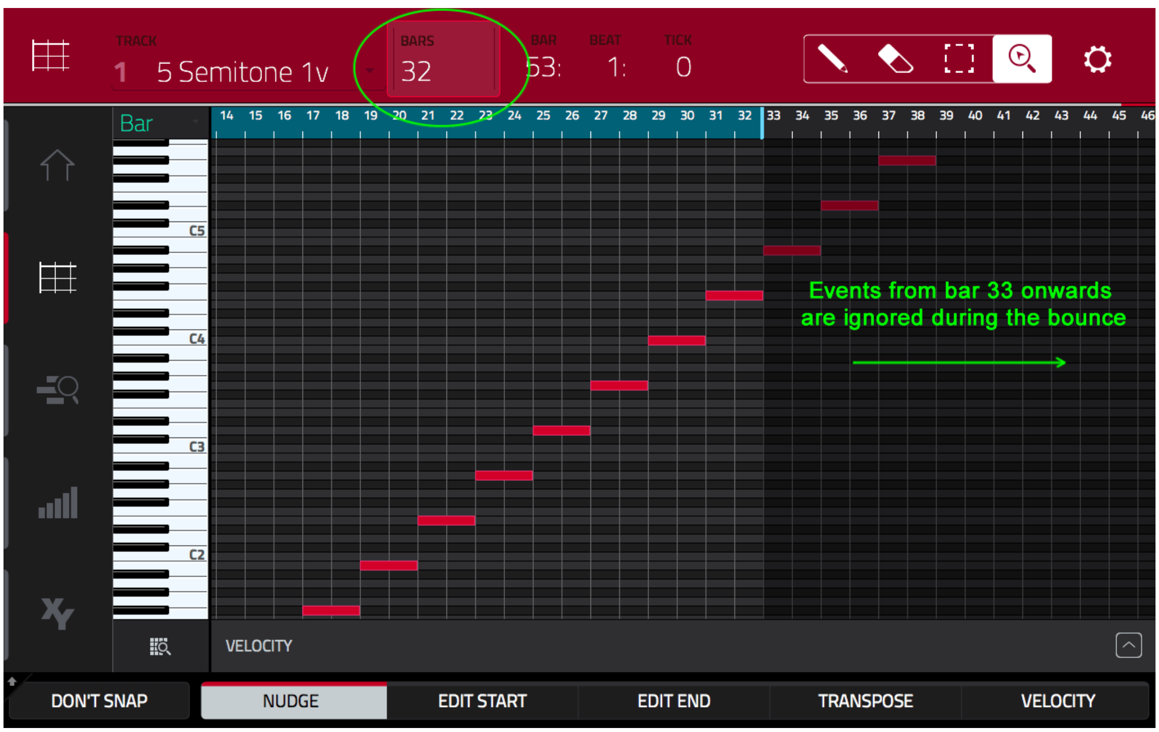
While you can just go ahead and create your bounced chops with the default sequence settings, each note is going to be sampled for 5 seconds which means you'll have a long empty gap at the end of each sample. This is not very efficient in terms of memory usage, and while we can edit the individual samples (either at the chop stage or after they've been chopped), it's a lot easier to adjust our sequence instead.

In **MAIN**, start to increase the **BPM** from the default **100.00**. If you increase the tempo to **200 BPM** (i.e. twice the tempo), you'll get notes that are half as long, i.e. 2.5 seconds.

Hit **PLAY** and listen carefully, we want to make sure the bass note has completely faded out before the next note plays. Sounds pretty good to me!

The next tweak you could perform is to remove those really high notes from the sequence. After all, this is a bass, not a lead sound, so do we really need all those squeaky notes in banks G and H?

I reckon the last event worth sampling is the one at bar 31, so in **GRID VIEW**, change **BARS** from 52 to **32**. As you can see, all the events from bar 33 onwards are now disabled and will not be included in our bounced track.



Now repeat the workflow exactly as you did before - the end result is a perfect clone of the original bass patch.

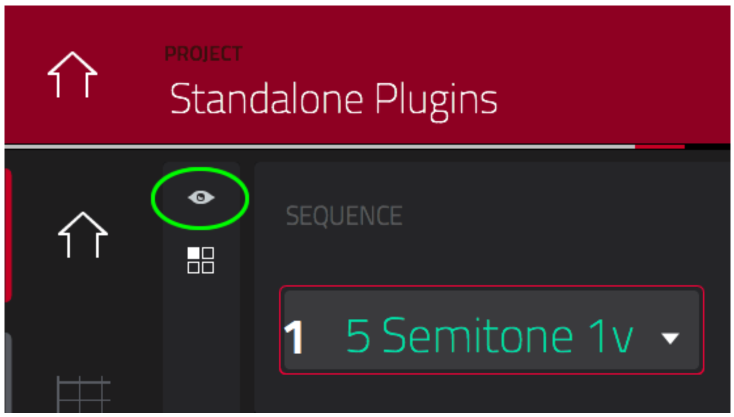
Tweaking Your Instrument Programs

The keygroup instruments you create are just standard MPC Keygroup programs, so you can tweak these to further customise your instruments.

Let's take a look at four easy tweaks you might want to make:

Adding Effects

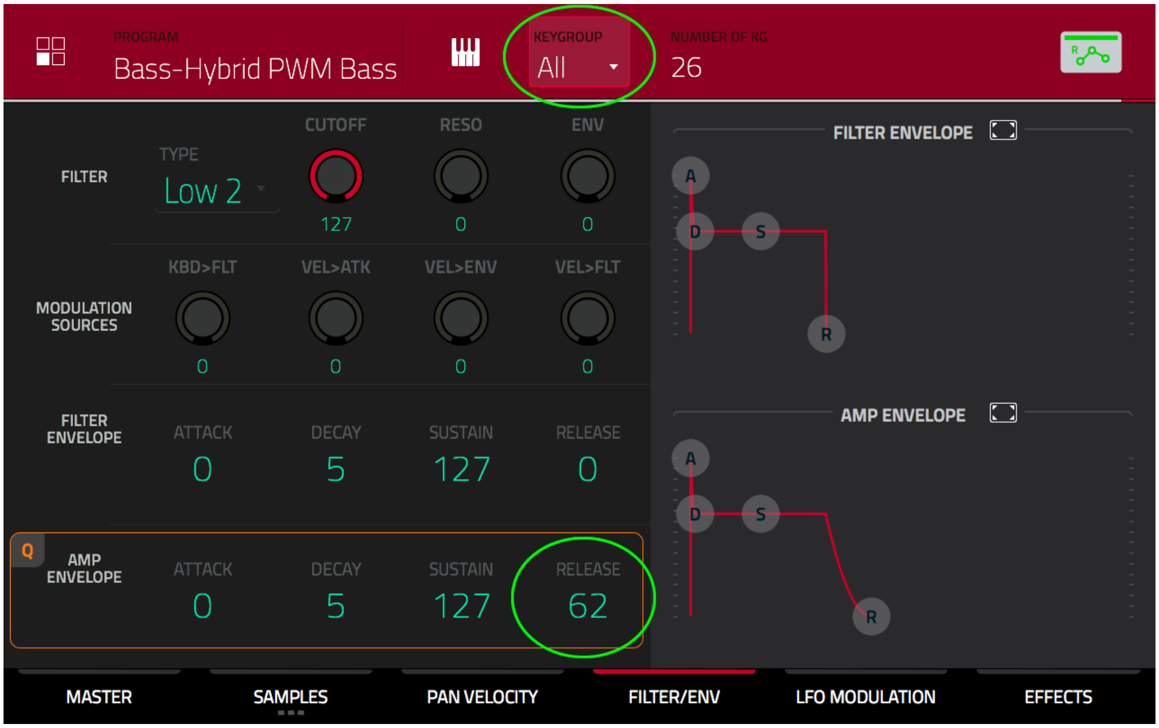
We previously disabled the effects from our plugin patch as ideally we want to be able to control this kind of thing at the program level. Adding effects to your program is very easy. Make sure your program is assigned to the currently active sequence track and click on the little 'eye' icon in MAIN:



This will reveal the Inspector panel. To add an effect to this program, tap in the INSERTS box to reveal the Inserts window. Here you can choose to add up to 4 insert effects. Remember to retain 'standalone' compatibility, you must choose Internal or Vintage Effects.

Adjusting The Release

The generic template has a fairly short release on all keygroups, but for some sounds you'll need something much longer, such as pads or strings; this way, when you 'release' the note you'll get a nice amount of gentle fade out. Go to **PROGRAM EDIT > FILTER/ENV**:



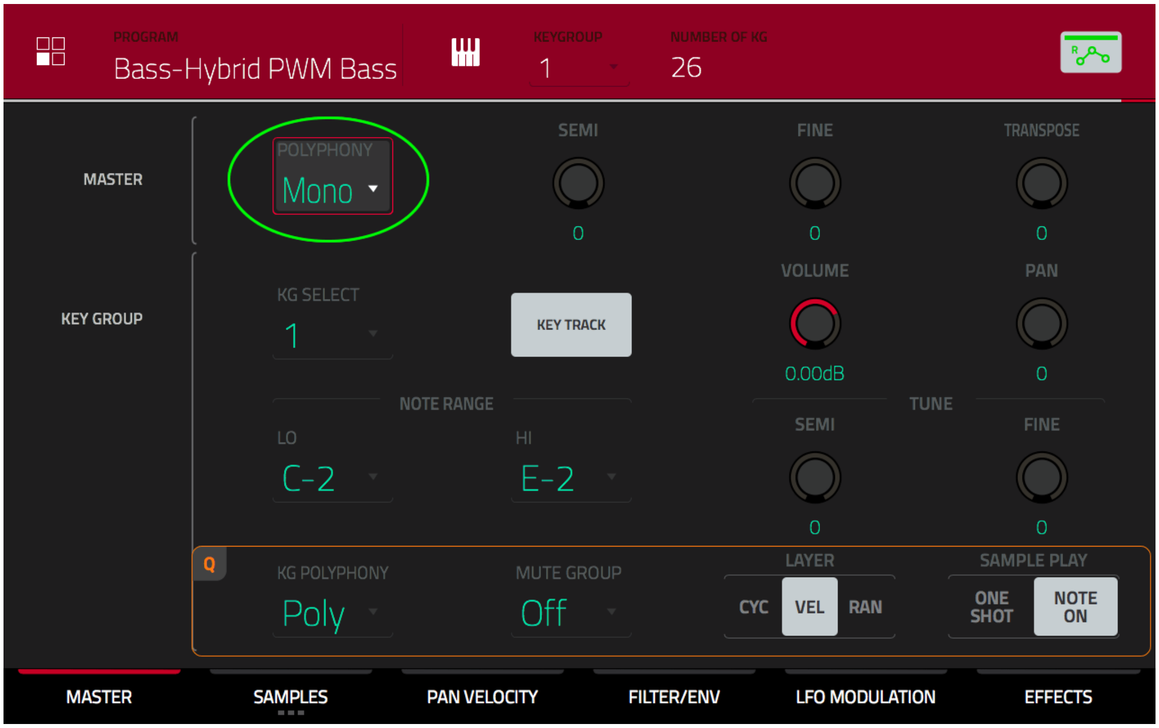
At the top of the page, tap on the '**KEYGROUP**' box and turn the data wheel anticlockwise to select '**All**' - now any changes you make to the envelope will apply to all keygroups.

Under AMP ENVELOPE, tap on the RELEASE value and adjust this to suit your sound - keep previewing the sound with your pads as you adjust the release.

Don't forget you can of course adjust the other envelope settings to further customise your instrument - for example increasing the attack value will produce a softer, more gentle start to each note.

Creating a Monophonic Instrument

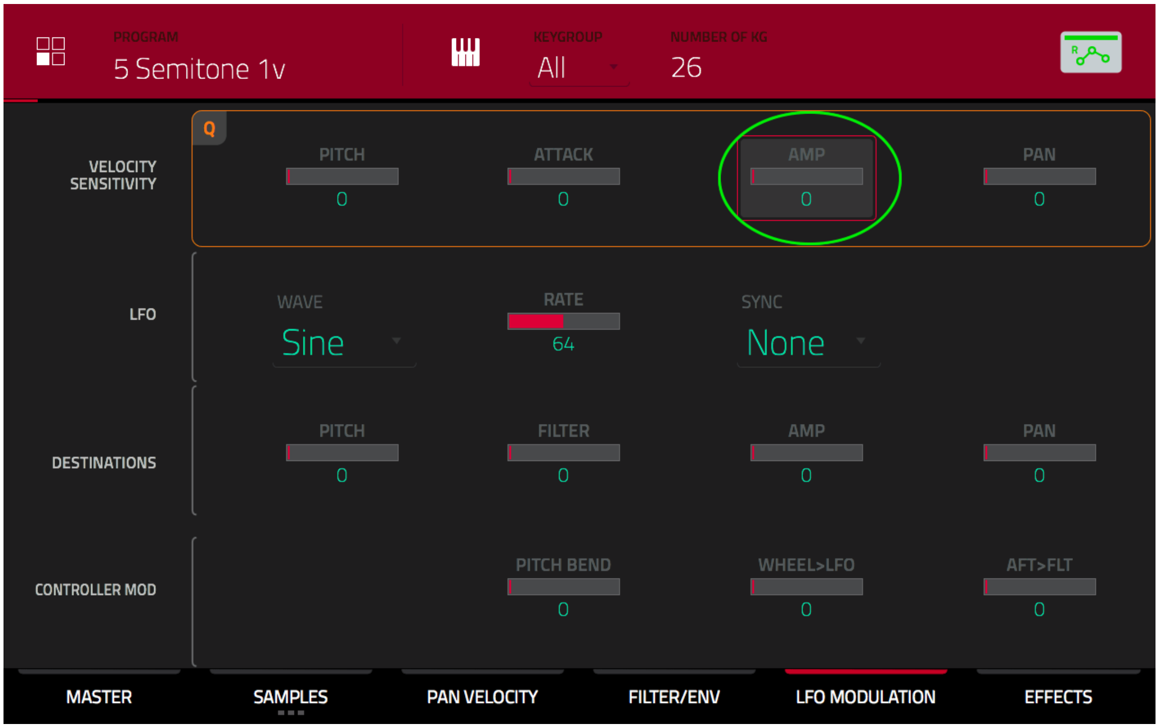
Many lead and bass patches benefit from being 'monophonic' which means they can only play a single note at a time. Go to **PROGRAM EDIT > MASTER** and change **POLYPHONY** from '**Poly**' to '**Mono**'.



Creating a Program with Zero Dynamic Range (Best for Organs)

Some instruments have no dynamic range at all, the most famous example of this is a Hammond B3 organ. Play at note at any velocity and it always outputs at the same volume, even if FULL LEVEL is turned off.

To mimic this behaviour, set **KEYGROUP** to **ALL** and in the **LFO MODULATION** page, go to the **VELOCITY SENSITIVITY** row and set **AMP** to **0**.

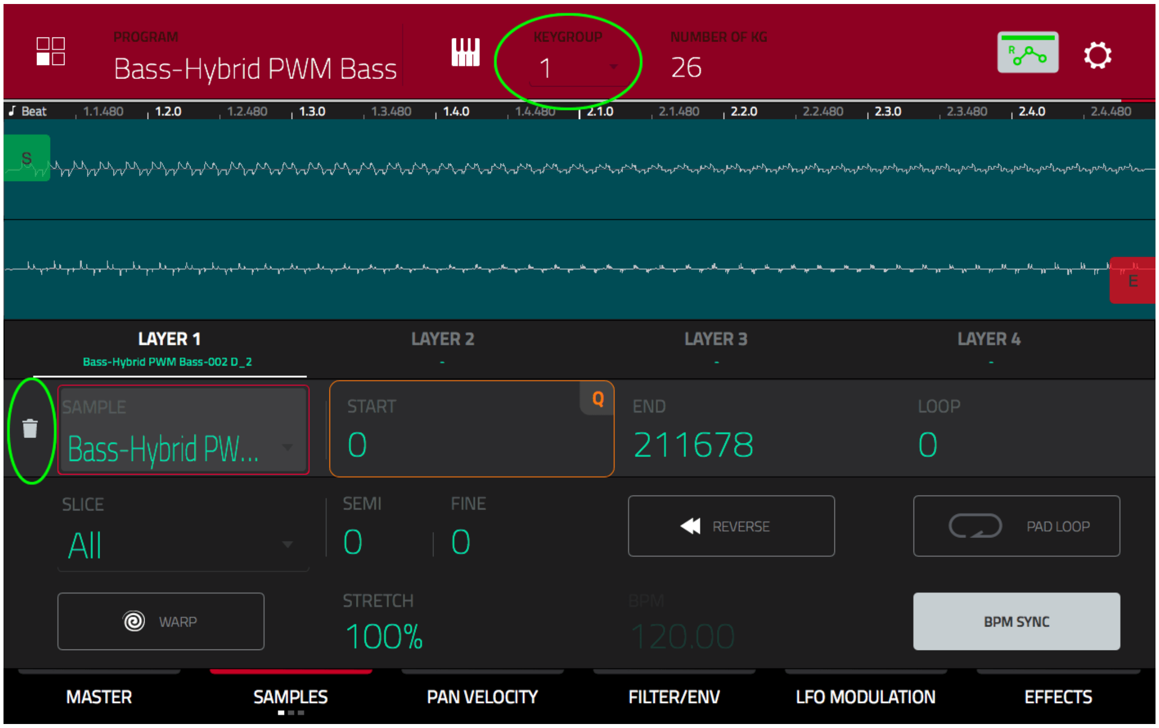


Removing Unwanted Samples

Before we bounced our bass patch sequence we reduced the number of bars to avoid recording the excessively high notes. What we couldn't do at that stage was remove the excessively low notes as this would have messed up the automatic slide labelling and assignment in the templates.

Now we can't actually fully 'delete' keygroups from a program, but we can remove the unwanted samples from the keygroups - if a keygroup has no samples, it effectively becomes 'disabled'.

Go to **PROGRAM EDIT > SAMPLES** and at the top of the screen select **KEYGROUP 1**.



In **Layer 1**, single tap the trash can icon to the left of the sample name to remove the sample assigned to this keygroup. Repeat for all the key groups you wish to disable - remember you can also do this for any higher range key groups you may wish to remove retrospectively.

Once you've removed all the samples, re-save the keygroup program - the program will now use up less memory when loaded into a project.

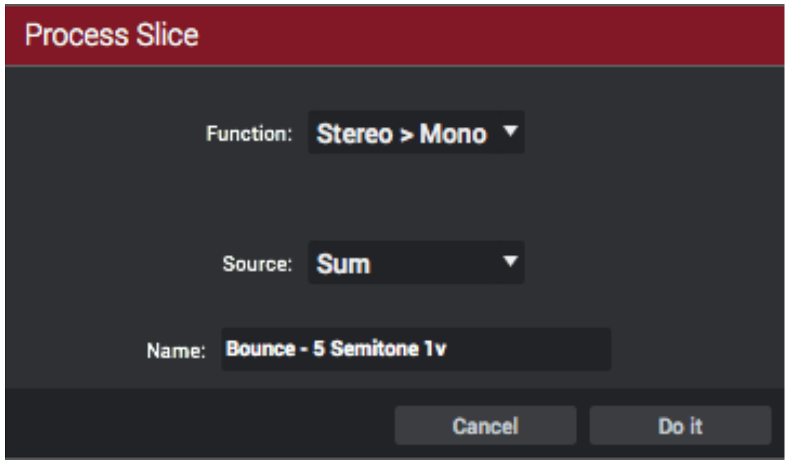
Don't be tempted to just reduce the number of keygroups in the program as this doesn't actually 'delete' anything - the keygroups (and the samples assigned to them) are still there, just hidden, so the samples will still be saved with the program.

Converting To Mono

The MPC always bounces sequences in stereo, but in order to save memory we can often get away with using mono samples in our instruments. You can of course convert your stereo 'chops' to mono at a later date, but it's easier to convert the entire bounce sample before chopping it.

After bouncing your track, go to SAMPLE EDIT and rename the original stereo bounce to something like '**Bounce - 5 Semitone 1v Stereo'.**

Now select '**PROCESS** > **STEREO TO MONO**'.



You can choose to merge both channels or use just left or right, the best option will vary depending on the patch so you might have to experiment.

Use the ‘**Name**’ field to give the converted mono sample the correct name for the workflow; '**Bounce - 5 Semitone 1v**’ (this is why I renamed the original stereo bounce so there wouldn’t be a name clash). Hit Do It and now region chop this new mono version of the bounce. As it has the correct name for the workflow, there will be no problems and all the resulting chops will be mono.

Stereo or Mono?

Due to the limited memory available in standalone mode, I definitely recommend sticking to mono samples whenever possible. Many sounds are 'naturally' mono, such as bass, guitar, lead sounds etc, and any stereo aspects are typically added later, such as reverbs, delays, chorus and so on.

If you are sampling something truly 'stereo' do also try making a mono version so you can compare the two and assess whether there is any obvious difference, especially after adding a stereo effect across the mono program. Also, don't forget the 'pseudo-stereo' effect I covered in chapter 22.

Program Previews

If you want to hear an audio preview of your new instrument each time it’s selected in the BROWSER, you’ll need to create a ‘**Program Preview**‘ for each one. I cover creating program previews in **Appendix C**.

The 3 Semitone Template

The '5 semitone' workflow we've been using tunes each sample up or down by a maximum of two semitones which in my opinion provides a good compromise between sound quality and memory usage, especially for typical synth sounds.

However, if you are looking at multisampling more complex plugins then you might want to consider limiting sample tuning to +/- one semitone, or even 'chromatic' sampling (where you sample every single note).

If you select **sequence 2** in the '**Standalone Plugins**' project you'll find the '**3 Semitone 1v**' sequence. Use this in the same way as the 5 Semitone sequence to generate '3 semitone' multisamples - however this time the default region chop should be set to **42 regions** as there are 42 events (84 bars, with each event covering 2 bars).

This will produce chops for the total note range of **C-2 to G8**, but remember that you can easily shorten the note range by reducing the number of bars via the BARS parameter - if you do, just remember to change the number of chop regions to match the actual number of events generated.

Once you've chopped your bounce, go to the '**Instrument Templates**' folder and load the '**3 Semitone 1v'** program and your '3 semitone' program is generated.

Just be aware that each time you increase the number of multisamples you increase the memory required for these programs so its important to remember convert your bounce to mono whenever possible.

The 4 Velocity Template

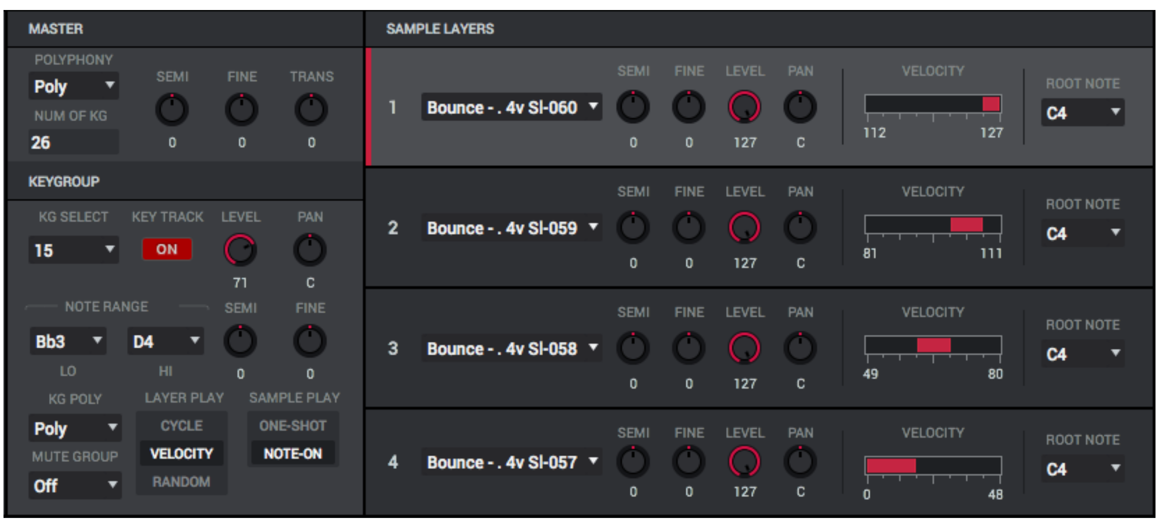
More dynamic instruments (e.g. pianos, guitars etc), where the sound and timbre changes depending on the velocity you strike the note, will typically require multisampling each note at multiple velocities.

As we’ve seen previously in this book, the MPC can support up to 4 velocity layers per pad/keygroup. Select **sequence 3** (**5 Semitone 4v**).

In this sequence I've placed events at every **5 semitones** but this time included **4 events for each note**, with each event set at a different velocity; **32**, **64**, **95**, **127**.

It works in the same way as the original '5 Semitone 1v' workflow, except this time the default region chop is **104 regions**.

After chopping, import the '**5 semitone 4v**' program template to automatically map all 104 regions to the correct key groups and layers.



In the screenshot above you can see that in the example keygroup (15), each layer has a root note set to C4 and has assigned slices 57 to 60 to the four layers. Each layer has a unique velocity range to control the dynamic switching.

4 Velocity Memory Concerns

To reduce the memory used by your 4 velocity program, it's vital that you only sample a 'usable' range', so before bouncing consider reducing the number of bars to automatically skip those ridiculously high pitch notes. And remember you can remove any unwanted low notes by unassigning samples from the key groups in the resulting clone program.

Also remember that before chopping a '4 velocity' bounce you should seriously consider converting the bounce to mono to halve the amount of memory used by your chops. Just remember to delete the original stereo bounce and rename your mono conversion to **Bounce - 5 Semitone 4v** before chopping it.

When Should I Multisample Multiple Velocities?

There's no escaping the fact that multisampling at 4 velocities is always going to use up a lot of memory, so before do so, spend a bit of time evaluating your plugin patch to decide whether you truly need to create so many samples. Ask yourself the following questions:

Are the softer velocities nothing but quieter versions of the higher velocities with no other difference in timbre?

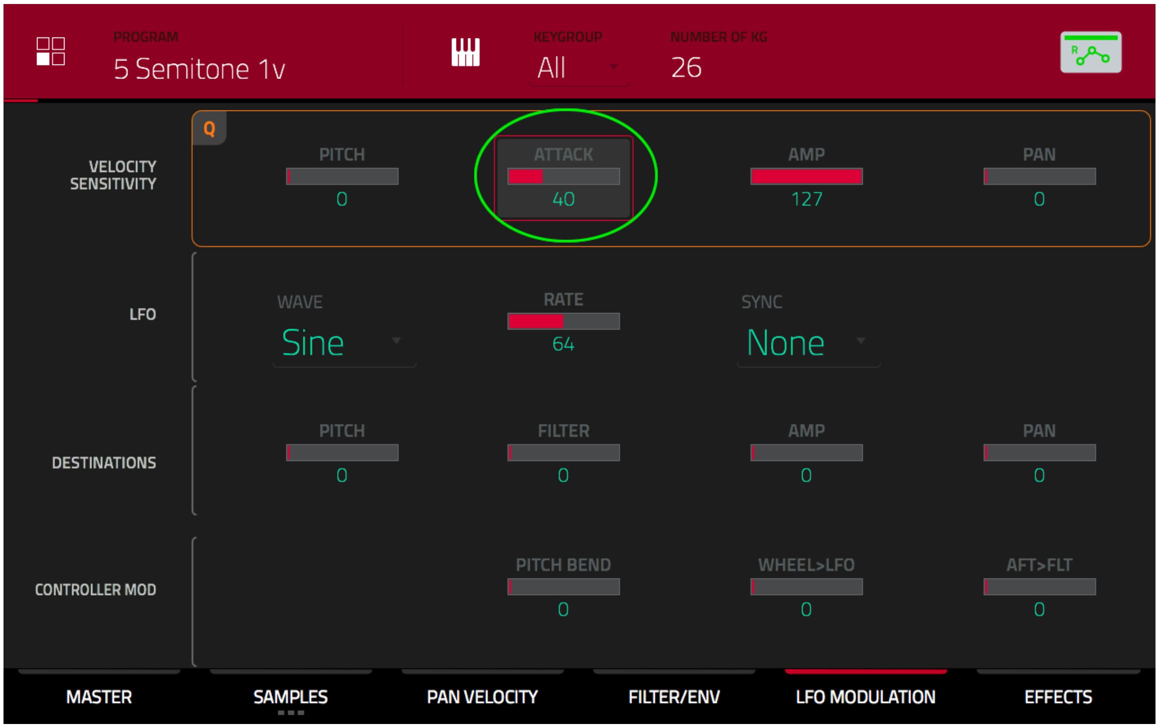
If so then stick to sampling a single velocity as your pads will automatically play the lower velocities more quietly, there's just no need to sample sounds that are just 'more quiet' than the 127 velocity version!

Are there timbre changes at lower velocities?

If those over velocities are definitely changing timbre in some way, then consider whether you can just sample the single velocity (at 127) and use velocity-sensitive program parameters to *replicate* those timbre changes.

We looked at velocity sensitive changes back in chapter 17 when we built a velocity sensitive drum kit, which are found in the **PROGRAM EDIT > LFO MODULATION** screen. These parameters allow us to change the way a program function acts depending on how hard we hit a pad.

For example, if the lower velocities are softer sounding we can configure a velocity sensitive envelope on a keygroup that decrease the attack value the harder we strike the pad. In the example below, lowest velocity hits will apply an **attack** of **40** on our keygroup; as we hit the pad harder, that attack value will decrease until it reaches 0 at a velocity of 127.



Just remember to apply your velocity sensitive settings to '**ALL**' key groups.

Example: Single Velocity With Velocity Sensitive Filtering

Let's look at a practical example of using velocity sensitive settings to avoid having to multisample 4 velocities. Load up the Standalone Plugins project and select sequence 1 (5 Semitone 1v).

Assign the **Hybrid** plugin to **track 1** and select the patch '**18 Hard Leads > 030 Moogish Lead**'. Select **BANK D** and make sure **FULL LEVEL** is '**off**'. Now preview some pads, varying the velocity from very soft all the way up to very hard.

You should hear that there is a filter acting throughout the entire velocity range with soft hits sounding very deep and hard hits sounding more bright and open. It certainly sounds like this patch could benefit from multisampling over multiple velocities, but instead let's see if we can emulate this sound with velocity sensitive filtering.

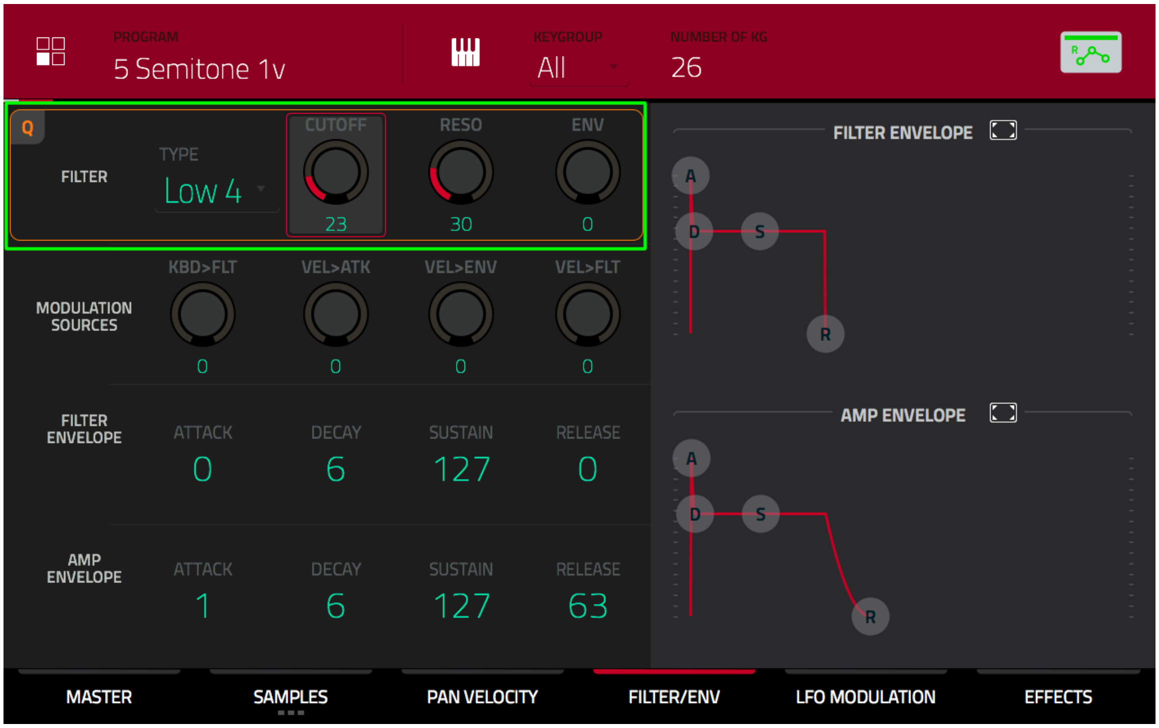
First, let's get rid of the delay effects. Open the **Hybrid plugin GUI** and select the **EFFECTS** section:

There's two sets of effects on this patch. In the top left (**PART A INSERT 1**) there is a **delay**, so click on the yellow '**ON**' switch to turn this **off**. Then in the **MASTER EFFECTS** section at the bottom, set the **CHORUS** and **REVERB** tabs to **OFF**.

Now perform the workflow on this patch to create your single velocity clone. Convert your bounce to mono before chopping to 26 regions.

After you've chopped and imported the **5 Semitone 1v** program template you'll have your single velocity clone. Play some of the pads at varying velocities to hear how it sounds. It's okay, but it doesn't have that dynamic filtering that the original patch had, so let's try to mimic this.

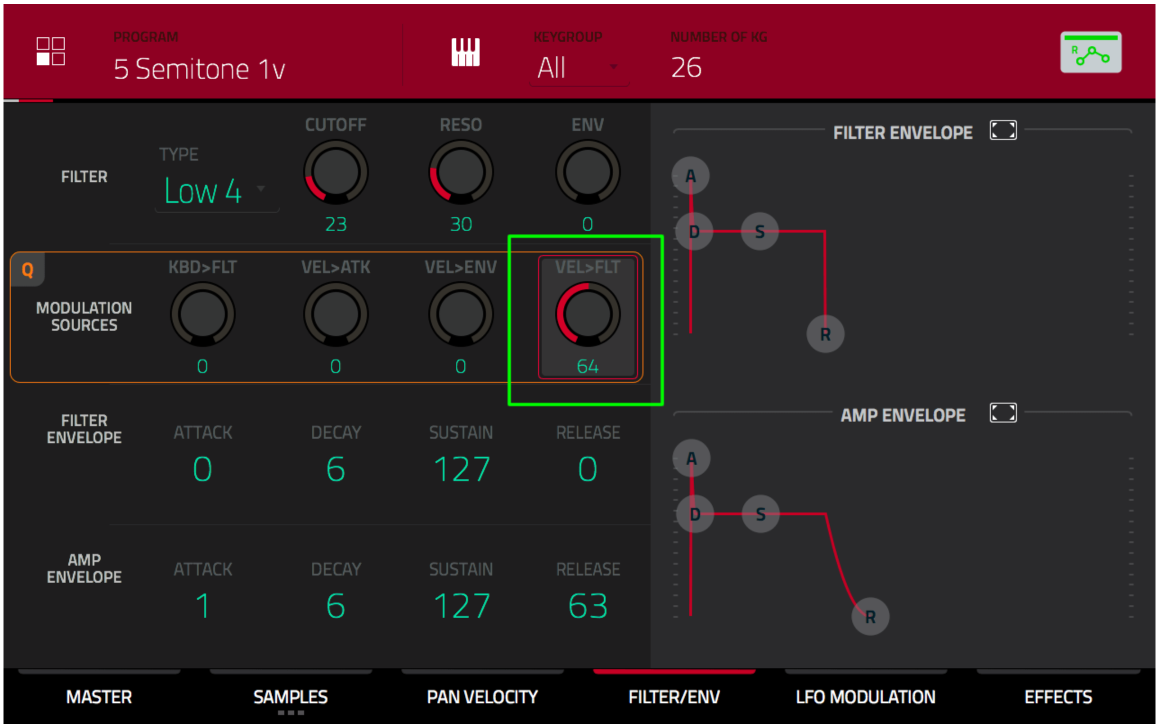
Select **ALL** key groups and go to **PROGRAM EDIT > FILTER/ENV** (you can do this in the hardware UI if you wish):



Try a **LOW 4** type filter like the one in the screenshot above. The low **CUTOFF** value gives us a very deep, filtered sound; I added a bit of resonance to give the filter when bite as it opens up.

Preview some pads and you'll hear the filter in action, but at the moment it's just heavily filtered the sound of the pad, there's no velocity sensitive settings.

Under **MODULATION SOURCES** begin to increase the **VEL>FILTER** setting. Preview a pad at varying velocities while you increase it. Try a value of **64**.



With VEL>FLT enabled our filter is now velocity sensitive, with deep filtering at low velocities and brighter, ore open filtering at higher velocities.

This is much more similar to the original patch - you can compare the two by switching the current track back and forth between KEYGROUP and PLUGIN type and previewing the pads. While not identical, it's pretty close and I'm sure you could experiment with the filter even further to get it even closer.

However it's actually pretty cool to create your own unique version of the patch - in fact you can of course use any filter settings that sound good to you (including multiple versions the use different filters).

and remember, when working with 4 velocity switches there's always a noticeable switch in sound when you cross one of the velocity 'boundaries', but you don't get this when using velocity sensitive parameters as the change in sound is seamless.

Inside the '**Saved Instruments**' folder you'll find my version of this clone inside the '**Moogish**' sub folder. Here I've added some delay and reverb and added a little more **VEL>FLT** to let the filter open up more at hard velocities. I've also removed some of the key groups so it covers a usable range of **C#0 to C#7**; the total instrument size is just over 8MB, which is pretty small for such a large sounding patch!

027 Final Words

I hope this book has given you a much deeper insight into what's possible from the MPC X and MPC Live! I'd love to hear your feedback on the book so if you have any ideas for improvements or if there's any specific tutorials you like to see me create in the future, please do get in touch via support@mpc-samples.com.

Happy beat making!

Andy Avgousti (MPC-Tutor)

Appendix A: Setting Up Your MPC

In this section, I wanted to give guidance on how to initially set up your MPC in preparation for the topics covered specifically in this book – quick and to the point!

Which Cables Should I Use?

There's a lot of different types of cables and socket types in the audio world so it can be quite confusing if you are just starting out!

Connector Types

The MPC Live and MPC X both accept two types of audio input/output connection types; **1/4 inch jack** and **RCA**, while the MPC X also accepts **XLR**.

RCA cables are used exclusively in your MPC for connecting turntables (see how to do this below). Turntables tend to use RCA cables, often with a built in ground wire.

RCA & Ground Cable

XLR tend to be used when connecting microphones into the MPC X's dedicated Mic inputs (more about this later).

XLR Connector

1/4 inch (6.35mm) jack is the main size connector used for both input and output connections on your MPC. There are two distinct types of 1/4 inch jack available.

TS jacks are often found on guitar cables; they are designed to carry an **unbalanced** mono signal:

TS 1/4 inch jack

TRS jacks (note the additional black ring) are able to carry **balanced** mono signals. A balanced signal tends to have less interference and hum so is perfect for pro audio applications.

TRS 1/4 inch jack

Your MPC has balanced inputs and outputs so as long as the device at the other end of the cable is also balanced, TRS cables will provide a fully balanced signal. If the other device is unbalanced then TRS cables will still carry a mono audio signal, but it will be unbalanced and potentially more prone to hum and interference.

Using Adapters

Many consumer audio devices use RCA connectors such as CD players, DAB radios and so on. If you wish to connect these to your MPC then you'll need to either use an RCA-to-jack adapter, or buy a cable with jack at one end, RCA at the other. Do not connect these devices to the MPC's RCA inputs as these are only intended for phono level audio sources (i.e. turntables).

RCA to TS 1/4" jack
cables & adapters

Many devices these days feature a 3.5mm mini-jack stereo output, such as smartphones, tablets, laptops etc. To connect these to your MPC, use a cable with a 3.5mm stereo mini-jack at one end that splits to two 1/4 inch TS jacks.

3.5mm stereo mini jack
to 2 x (TS 1/4" jack)

MPC Outgoing Audio Connections

To hear the audio from your MPC, connect **1/4 inch jack** cables to the **L and R** '**OUT**' ports at the back of your MPC. Now connect the other ends of those jacks to your studio's sound system. This could be direct to your amp (or active speakers):

Or via a mixer, in which case you can take advantage of the other outputs

You can also use headphones – just connect them to the dedicated headphone port at the back of the MPC via a 3.5mm stereo mini jack connector and use the ‘**VOL**’ dial to control the output volume.

Incoming Audio Connections

There are a few different types of audio sources you will typically record:

- **Line level** – this is the audio from CD players, iPods, smartphones, DVD, DJ mixers, other samplers etc
- **Mic level** – the signal received from microphones.
- **Instrument Level** – the signal from electric guitars and some keyboards
- **Phono Level** – the signal directly out of the back of a traditional turntable

Signals going into your MPC eventually need to be raised to line level; this is either done before the signal reaches your MPC, or in some instances can be done by the MPC itself.

Line Level Devices

MPC X:
Connect **line level** devices directly to **inputs 1 and 2** using **1/4 inch jack cables**. You can also connect line level sources to **inputs 3 & 4** using either 1/4 inch jack or RCA cables. Remember to set the switch to **LINE**.

MPC Live
Connect **line level** devices directly to **inputs 1 and 2** using either **1/4 inch jack** or **RCA** cables. Remember to set the switch to **LINE**.

Remember: *If you are running a turntable through a DJ mixer, the output will be line level.*

Microphones

MPC X:
Connect all **microphones** to **inputs 1 & 2** using either **1/4 inch jack** or **XLR** cables. Condenser mics will require 48v phantom power so make sure you switch this on via the top panel of the MPC X.

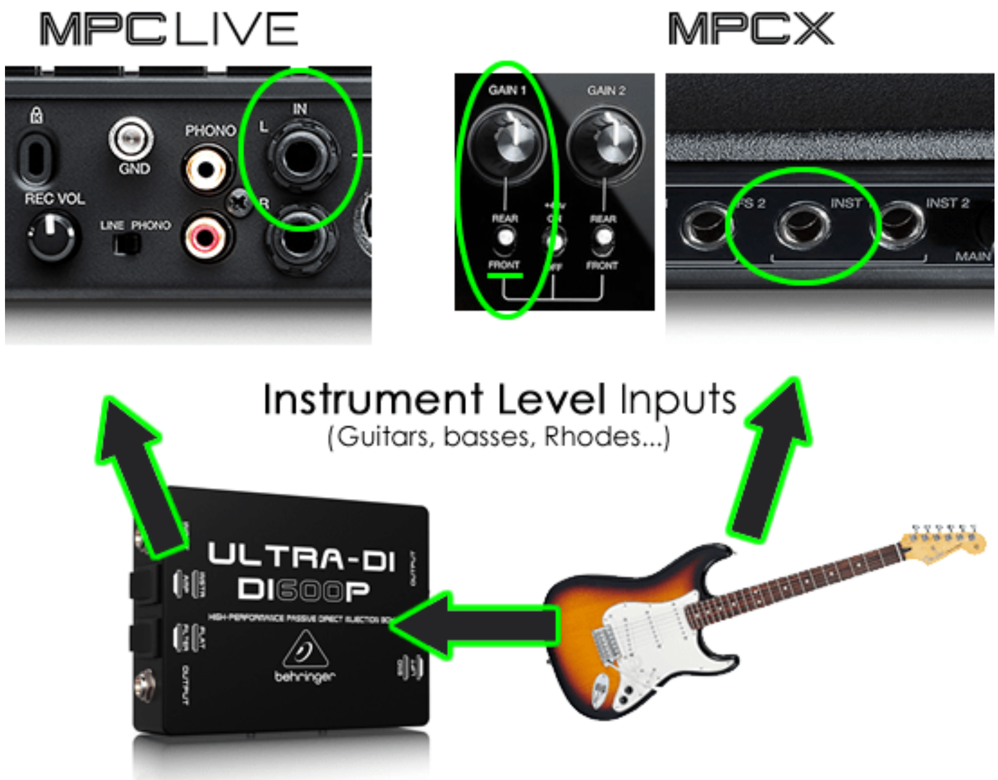
MPC Live:
Connect your microphone to a suitable **mic preamp** and then connect the preamp to the 1/4 inch jack inputs. If you are using a condenser mic, engage the 48v phantom power switch on your mic preamp. Please note that many mixers have mic preamps built in so you could run your mic via the mixer.

Turntables (Phono)

If your turntable outputs its signal at **phono** level, connect the outputs to dedicated **RCA inputs** at the back of your MPC (**inputs 3/4** on the MPC X) and set the switch to **PHONO**. If it has one, remember to connect your turntable's 'ground' wire to the MPC's ground screw.

Remember: *If you are connected via a DJ mixer, then use the 'line' level instructions as the mixer has already raised the phono signal to line level.*

Instruments



MPC X:
Connect ‘**instrument level**’ sources (guitars, basses and some keyboards such as a Rhodes or B3 Organ etc) to the dedicated **‘Instrument’ inputs** at the front of the MPC X. Remember to set the **GAIN** controls on the top panel to **FRONT**.

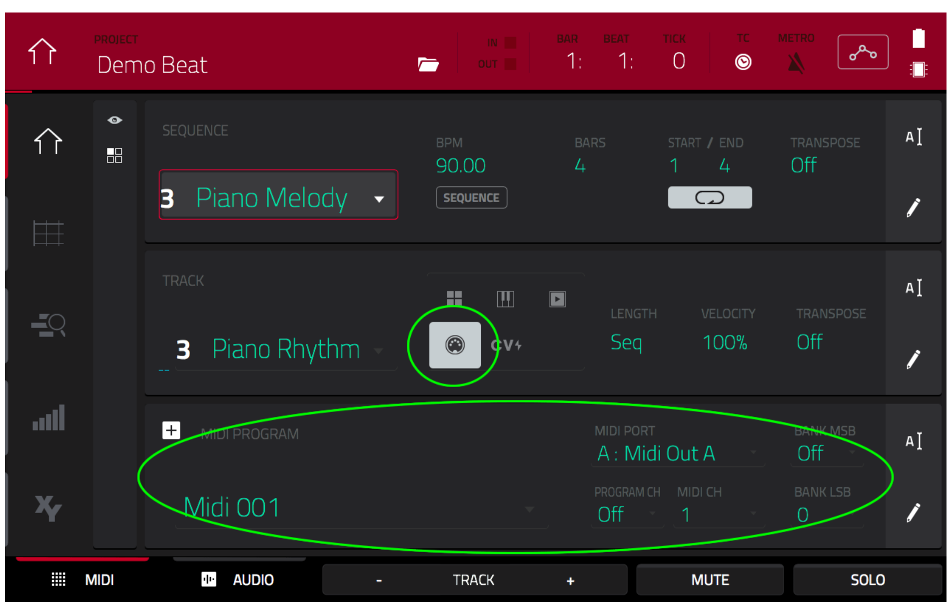
MPC Live:
‘Instrument level’ sources should be connected to a DI Box to bring the instrument level up to line level. Alternatively you can connect them to a dedicated amplifier (e.g. guitar amp) and either record the audio via a microphone, or use the amp’s own ‘line out’ (if it has one).

MIDI Connections

MIDI Sound modules

You can connect ‘old school’ MIDI instruments via the MPC’s physical MIDI ports using a standard **5 pin MIDI cable**. **MIDI OUT** of your MPC goes to the **MIDI IN** of your MIDI instrument. This way the outgoing MIDI ‘instructions’ from the MPC sequencer are received by the instrument.

In your MPC sequence, set track **TYPE** to **MIDI**, set **MIDI PORT** to match the MIDI port you connected to the instrument (i.e. A, B, etc), and match the **MIDI CH** number to match the MIDI channel you assigned to the patch in your sound module (consult your sound module's manual for its specific set up instructions)



Connecting MIDI Controllers

You can play KEYGROUP instruments using an external control keyboard. For 'class compliant' **USB keyboards** you just plug into a spare USB slot on your MPC (most keyboards should be supported), while for control keyboards with a traditional **5 pin MIDI 'out'**, simply connect this to any **MIDI IN** of the MPC using a standard MIDI cable. You do not need to configure anything inside the MPC for this to work.

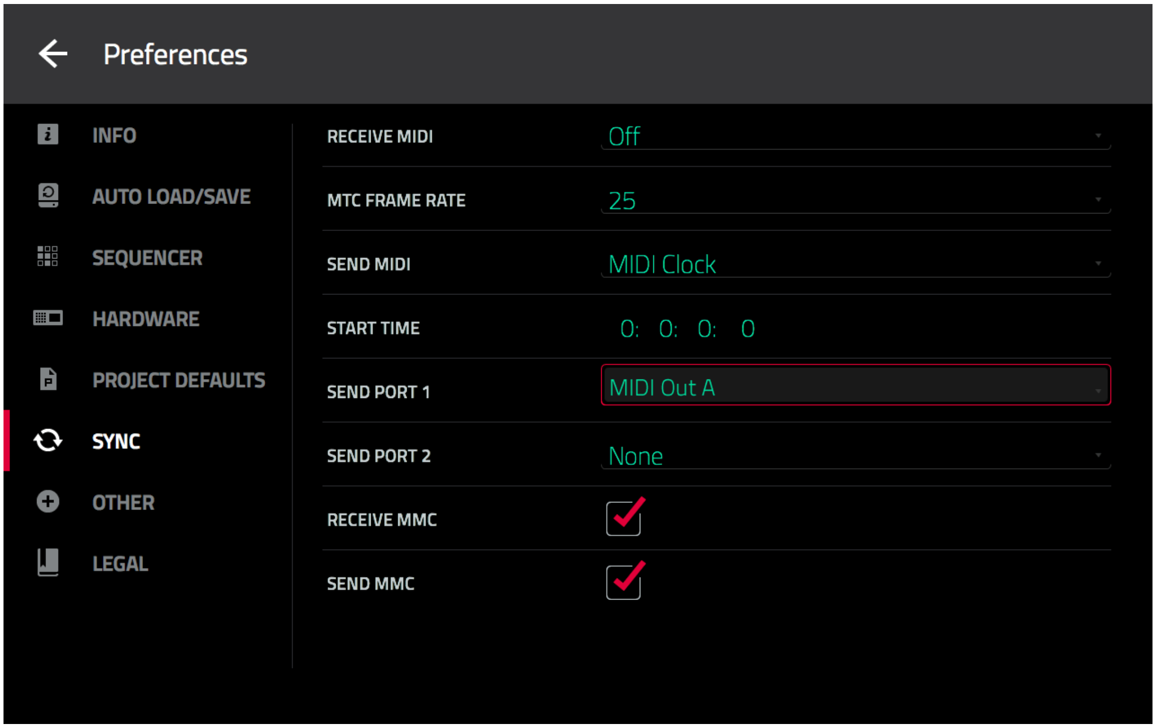
MIDI Sync

If you are syncing your MPC with another hardware MIDI device such as a multitrack recorder or hardware sequencer, first connect the MPC and the device in a 'MIDI handshake' so both devices can send and receive data from each other.

This requires **MIDI OUT** of your **MPC** into **MIDI IN** of your **other device** (so the MPC can send info to the device) and then **MIDI OUT** of the **device** into **MPC MIDI IN** (so the device can send info to the MPC).

Now go to **PREFERENCES > SYNC** to configure the required MIDI SYNC settings depending on the sync protocol used (MIDI Clock or MIDI Time Code) and which device is the 'master' (the one that 'sends' MIDI sync information) and which is the 'slave' (the one that 'receives' the MIDI sync information). Activate MMC to send 'transport control' messages in addition to standard MIDI sync.

Here's example settings where the MPC is acting as a MIDI clock 'master':



Software Installation

If you want to work in 'Controller Mode' you'll need to install the MPC Software application on your computer. Rather than re-invent the wheel, I'll point you to this video from **Noterepeat** that explains everything in 4 minutes, including how to activate your software using iLok:

https://www.youtube.com/watch?v=qnNGP4cEDuE

Appendix B: MPC File Transfers

While your MPC can be operated entirely independently of any computer, it's inevitable that at some point you'll need to connect to your computer for the purposes of file transfers, such as:

- Working on 'standalone' projects in the MPC Software computer GUI
- Making backups of your projects
- Transferring stereo masters and track exports to DAWs and other applications
- Transferring sample libraries and computer GUI projects to your MPC for use in 'standalone' mode

File Transfer Restrictions (Internal 16GB Drive)

Your MPC ships with a built-in 16GB internal drive which contains Akai's free sound library including the 'Vault 2' expansion; it also contains a folder called 'MPC Documents' which includes ready-made sub folders to store your own content.

It's very important to realise that while you can save content to this drive, Akai have provided no way of 'externally' accessing the content you save to it! This means it is incredibly tricky to transfer your 'standalone' projects to your computer; there's no easy way to back up your entire drive and it's convoluted (but not impossible) getting individual projects from this internal 16GB drive over to your computer if you would like to work on them in the computer version of the MPC Software.

So generally speaking I advise you consider the internal 16GB drive as a 'read-only' drive that is used to store the Akai sound content and instead save your work to an **'attached disk'**.

What is an 'Attached Disk?

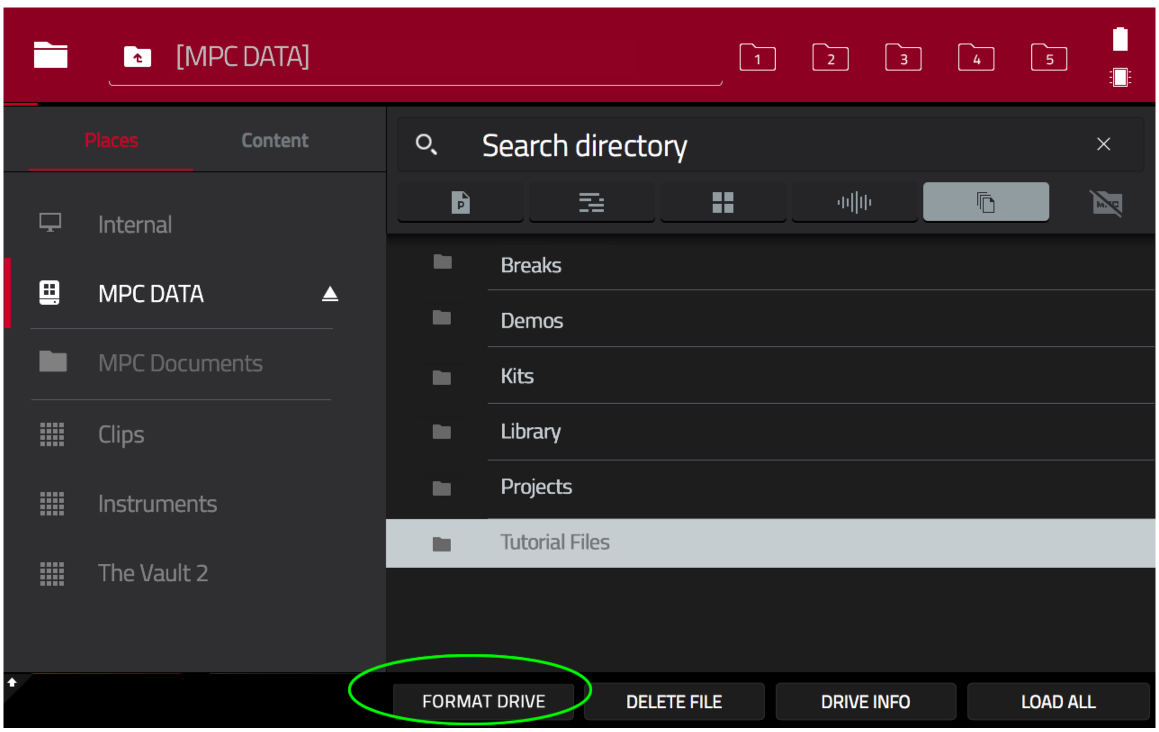
An 'attached disk' is any disk that is directly connected to your MPC and subsequently recognised in the BROWSER. This can be:

- An **SD card** connected to the SD card slot at the back of the unit; I'd advise you buy a good quality branded 'class 10' SD card.

- A **USB flash disk** or **USB portable hard drive** connected to any of the USB ports; for maximum portability I would suggest a low profile USB 3.0 disk as these only protrude the USB port by approximately 10mm.
- A **SATA disk** installed inside the MPC via the provided access port underneath your unit; an **SSD drive** is the best option as they are fast, noiseless, robust and (for Live owners) use less battery than a mechanical drive.

Before a drive can be used in your MPC it should be formatted to a recognised disk format. The MPC can read all common disk formats and it's likely that your disk is already formatted 'out of the box'. For removable disks (SD and USB) I'd recommend **exFAT** format (no file size restrictions and compatible with all computer OS), for an internal SATA I'd also recommend exFAT, but Windows users could also use you NTFS (Mac users will find that HFS+ formatted drives are 'read only' so this type of format is not recommended).

The MPC can also format disk from within the **BROWSER**. Simply tap on the disk you wish to format, hold down **SHIFT** and press the **FORMAT DRIVE** button:



The MPC will format the disk in exFAT format. It's worth nothing that if your disk is not formatted at all (common with some SATA drives) then the MPC will not be able to even detect the existence of this disk, so internal formatting would not possible. In this case you'll have to format on your computer - if it's a SATA disk you'll probably need to pop it inside a USB hard drive 'caddy' to connect it.

Whichever formatting method you choose, I would recommend you create some sub folders on the disk to help organise the content you intend to store on it. For example, '**Projects**', '**Exports**', '**Sound Library**', and so on. The 'Sound Library' folder could be further categorised into '**Drum Kits**', '**Expansions**', '**Loops**', '**Instruments**' etc. Unfortunately the MPC BROWSER doesn't currently support creation of folders, so you'll need to do this on your computer, either by connecting the drive to the computer directly, or via MPC 'controller' mode (more about this in the next section of this article).

Remember you can connect multiple drives to your MPC, so there is nothing stopping you having an SD card, two USB drives and an internal SATA disk connected simultaneously.

File Transfer Methods

This book is supplied with a folder of files that give you the opportunity to recreate all the tutorials in the book, so let's see how we can easily transfer this folder to your MPC.

After downloading and extracting the 'zip' archive for this book you should see a folder called '**Tutorial Files**'. We can transfer this folder to your MPC by two distinct methods.

Method 1: Direct Disk Transfer

If you are using a 'removable' drive in your MPC (i.e. SD card or USB flash drive), you can simply pull the disk out of the MPC drive and insert the disk into your computer. From here, just drag and drop your folders and files from your computer over to the disk. You can also copy folders and files from the MPC disk to your computer using the same drag and drop methods.

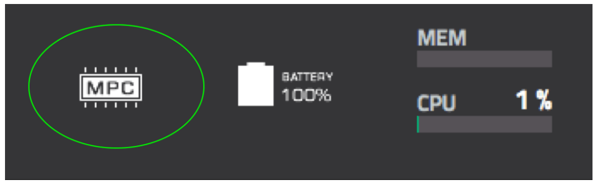
Once completed, eject the disk from your computer and place the disk back in your MPC.

Method 2: Controller Mode

If you are using an internal SATA drive, or simply prefer to leave your removable disks in your MPC, then you should use 'Controller Mode' to perform your file transfers.

Firstly, connect the 'square' USB port on your MPC to a spare USB port on your computer using a standard USB cable (a blue one is provided with your MPC although some people have had problems connecting with this one).

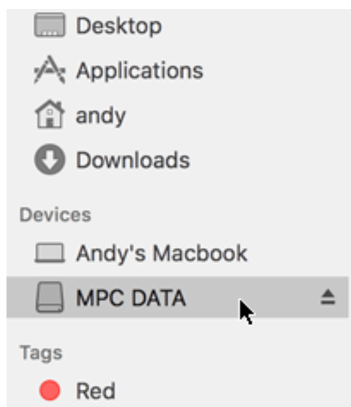
From the **MENU** screen, click on the '**MPC Chip**' icon at the top of the page:



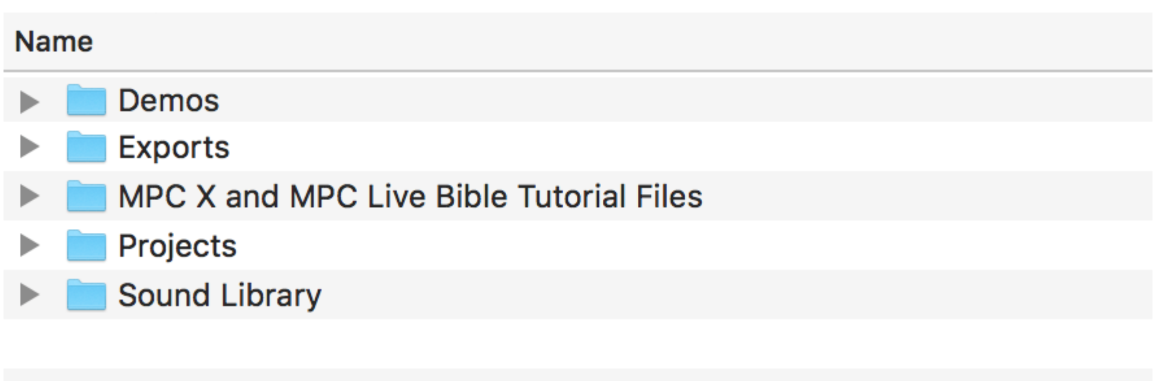
This will bring up the '**ENTER CONTROLLER MODE**' screen. Click on the **CONTROLLER MODE** button at the bottom of the page.

The MPC will now look for your computer. If the MPC Software application is open on your computer, the MPC will enter 'controller mode' which will allow you to control the computer MPC Software from your MPC hardware.

However you do not need the MPC Software to be open in order to access your MPC disk(s); after pressing '**CONTROLLER MODE**' all your attached MPC disks should appear as 'removable' drives in your computer's file manager (e.g. 'Finder' on a Mac, 'Explorer' on a PC).



Click on the drive to open its file system:



You can now drag and drop files to and from this disk. After copying your files and folders, make sure you safely eject the drive(s) from your computer before disconnecting your MPC from the computer to avoid any data loss.

Accessing Projects Saved to the Internal Drive

If you have projects saved to the internal drive you cannot access them directly in controller mode. The workaround is,

- While in 'standalone' mode, load the project you wish to transfer into your MPC memory,
- go to **MAIN**
- Tap the **PROJECT folder icon** at the top centre of the screen
- On the resulting **PROJECT** screen**,** hit **SAVE AS**
- Choose a save location on an 'Attached Disk' and hit **DO IT** – you'll now have a copy of your project saved to this additional location.
- If you have multiple projects that you wish to copy, you'll need to repeat this entire procedure for each one individually.

You can now connect the attached disk to your computer (either directly or via controller mode) and copy those projects to your computer.

How Do I Continue Working on a 'Standalone' Project in Controller Mode?

When it comes to storage and memory, the MPC sees 'standalone' and 'controller' modes as two very distinct and separate environments, so it is currently not possible to 'seamlessly' switch from the currently active 'standalone' project to continue working on it in controller mode.

Instead, the process is as follows:

- Save the current project to an attached disk in 'standalone mode'
- Connect your MPC to your computer via USB
- Launch the MPC Software on your computer.
- In your MPC, switch to 'controller' mode – your MPC disks should now appear as removable drives in your computer
- In the MPC Software BROWSER, navigate to your MPC's drive and load the project directly from the MPC disk

You could alternatively copy the project from your MPC disk to a preferred location on your computer and load that copy into the MPC Software application – however the above procedure ensures that the copy on your MPC disk is always the most current.

How Do I Backup My MPC Projects and Files?

The problem with working in ‘standalone’ mode is the MPC currently does not feature any kind of automated ‘system’ backup to a separate disk, so if your MPC disk becomes faulty or dies, you could permanently lose all your work!

To backup your MPC disk, first connect your disk to your computer using either of the methods discussed in this chapter. Once it appears as a removable drive on your computer you can copy your project folders or even the entire disk to a designated backup location on your computer.

Many third party applications offer a way to automatically back up a disk as soon as it is attached to your system so this would ensure your backups are made in the background each time you connect via controller mode.

Also consider backing up to a ‘cloud storage’ folder such as Dropbox, iCloud or Google Drive – this way you have an ‘off site’ backup in addition to your local backup.

Appendix C: MPC Expansion Guide

Throughout this book we've used the hardware BROWSER to load files into our MPC projects, but if you use the MPC Software computer GUI you are probably also familiar with the dedicated Expansion Browser and the Media Browser which are specifically designed for browsing MPC 'Expansion' packs.

Since the firmware 2.1 update, the touchscreen hardware BROWSER has now been given the ability to load MPC Software Expansions, albeit in a somewhat limited fashion compared to the computer GUI, but this small addition has proven to be very popular.

In this appendix we're going to look at how to create our own expansions from your own kits, instruments and sequences and how to export (and use) them in the 'standalone' browser.

What Is an Expansion Pack?

There are actually two types of Expansion pack; the first is a 'plugin expansion' which is a software instrument and is simply Akai's own version of a VST or AU plugin. The MPC Software has the Hybrid expansion pre-installed, but you can also download other plugin expansions from your akaipro.com account such as the Bank, Wub and Noise. Plugin expansions are not compatible with the MPC X and MPC Live in standalone mode.

The second type of expansion are 'sample expansions' and these will be the focus of this appendix. Sample expansions are collections of samples, programs and sequences, often based on a particular 'theme'. Your MPC comes with two sample expansions pre-installed in standalone mode; the '**Vault 2**' and '**Essential Instruments 2**'. You can also install expansions into the MPC Software computer GUI; there are several expansions available for download from your Akaipro.com account, plus you can purchase expansions from my site, MP-Samples.com: http://www.mpc-samples.com/section.php/79/0/mpc-x-mpc-live-tutorials-sounds-expansions/

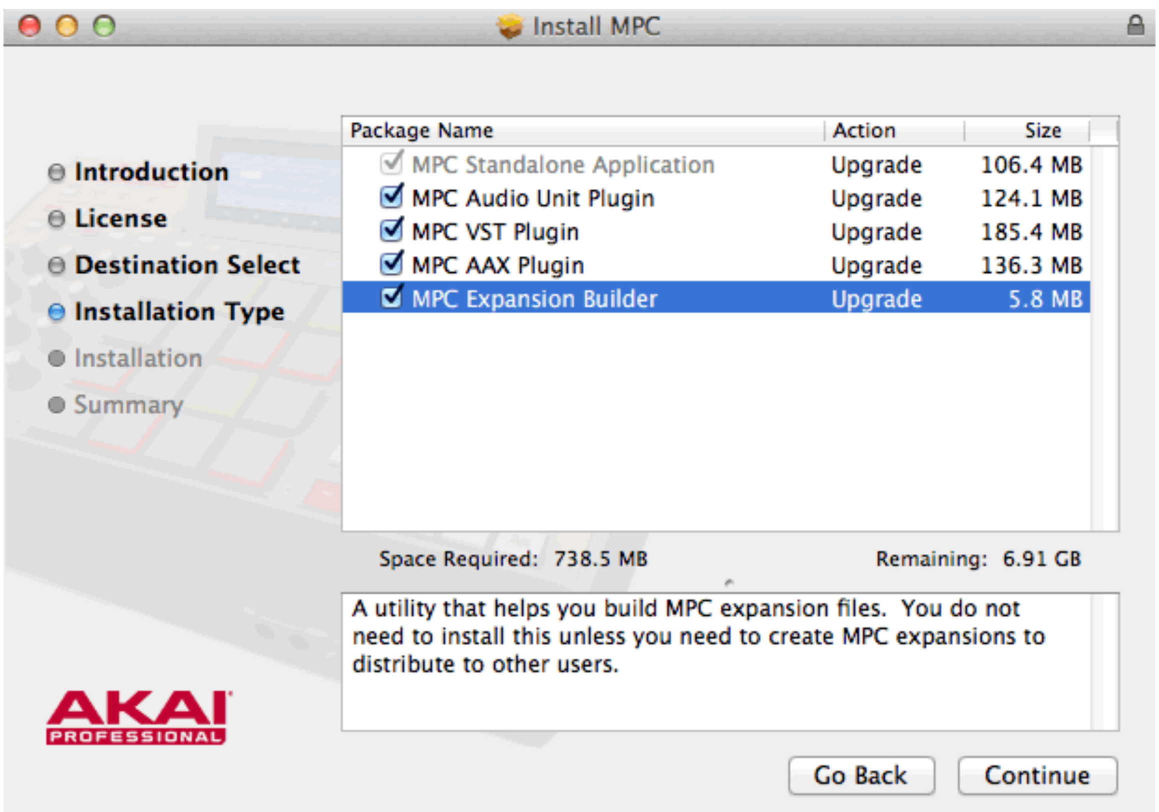
Later in this tutorial I'll show you how to load files from your expansions, both in the computer GUI and the hardware UI, but first let's now look at creating our own expansion pack and then install it for controller and standalone mode use.

Creating Your Own Sample Expansion Packs

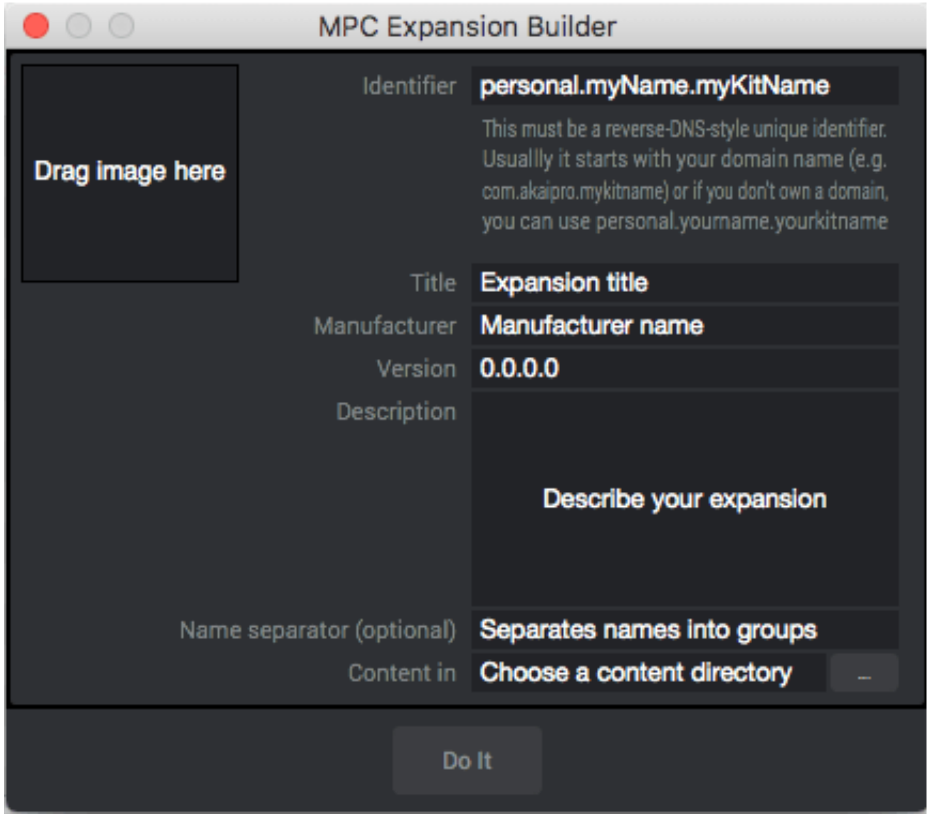
You can make your own sample expansions very easily using the separate '**MPC Expansion builder**' application that is bundled with the MPC Software installer.

On a Mac you'll find the expansion builder application in **Applications > Utilities > MPC Expansion Builder**. In Windows, you should find a shortcut in your **Start Menu** under **All Programs > Akai Pro > MPC Expansion Builder**. If there's no shortcut there, you can access it directly in **C:\Program Files\Akai Pro\MPC\ExpansionBuilder.exe**.

If the MPC Expansion Builder application is not installed, download the latest MPC Software installer from the Akaipro.com web site and launch the installer. Select '**Reinstall MPC Software**' and after agreeing to the License terms you should see the following page:



Make sure the **MPC Expansion Builder** package is checked and continue with the installation. After the installation is complete, launch the **Expansion Builder**:



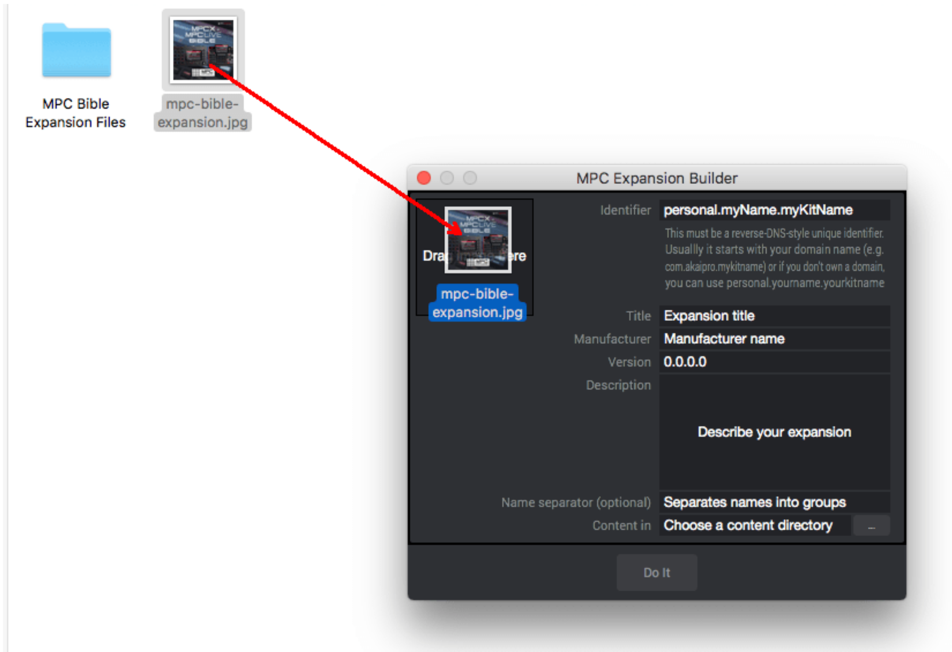
The concept behind building an expansion pack is simple – point the Expansion Builder to a folder containing the samples, programs and sequences you wish to make up the expansion pack (this is the '**Content Directory'**) and the expansion builder takes care of the rest, creating a single 'Expansion Installer' file (**XPN**) from the content it finds inside the folder.

You can see that there are a few other fields to complete so we'll take a look at these one step at a time. But the most important thing we need is some actual content to convert into an expansion pack!

Open the **Appendix B** folder and inside there you'll see a sub folder called '**MPC Bible Expansion Files**'. Inside this folder is a collection of sounds and programs that we'll use to create our expansion pack.

Also in the **Appendix C** folder you'll see an image file, '**mpc-bible-expansion.jpg'.** This is the image we'll display in the Expansion Browser to identify this expansion pack. When you create your own pack you'll need to use your favourite image creation software to make a perfectly **square** image; it doesn't have to be a specific size as Akai seem to use various sizes, although the latest expansion packs use images that are **1000 x 1000 pixels** in size, so I would suggest sticking to those dimensions if you can. Also it seems both **JPEG** and **PNG** format are accepted.

So to start making our '**MPC Bible Expansion'**, drag the provided image from your computer's file system (via Finder or Explorer) directly into where it states '**Drag Image Here'** in the Expansion Builder.



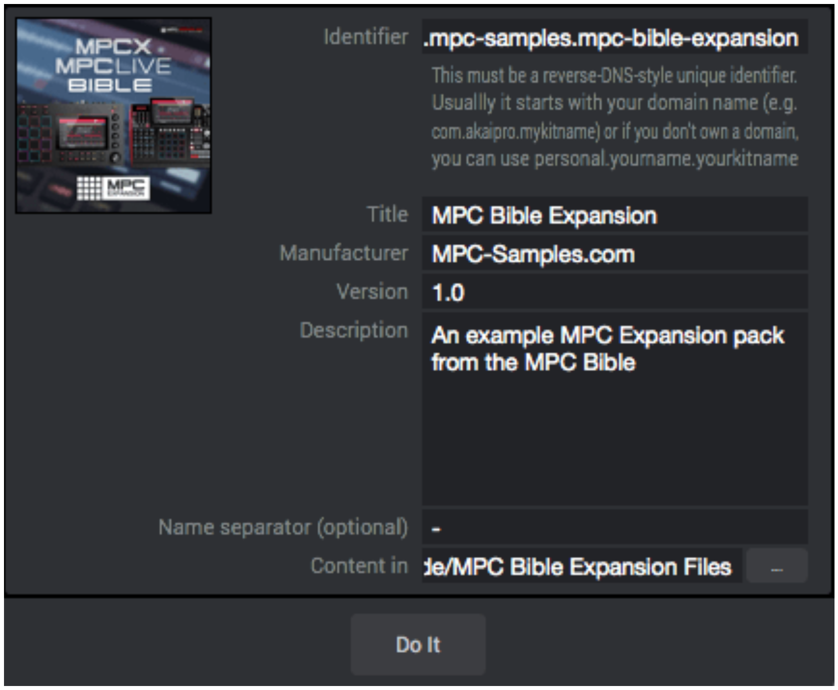
Now enter the following:

Title: **MPC Bible Expansion**
Manufacturer: **MPC-Samples.com**
Version: **1.0**
Description: **An example MPC Expansion pack from the MPC Bible**
Identifier: **com.mpc-samples.mpc-bible-expansion**
Name Separator: **-** (i.e. a 'hyphen')

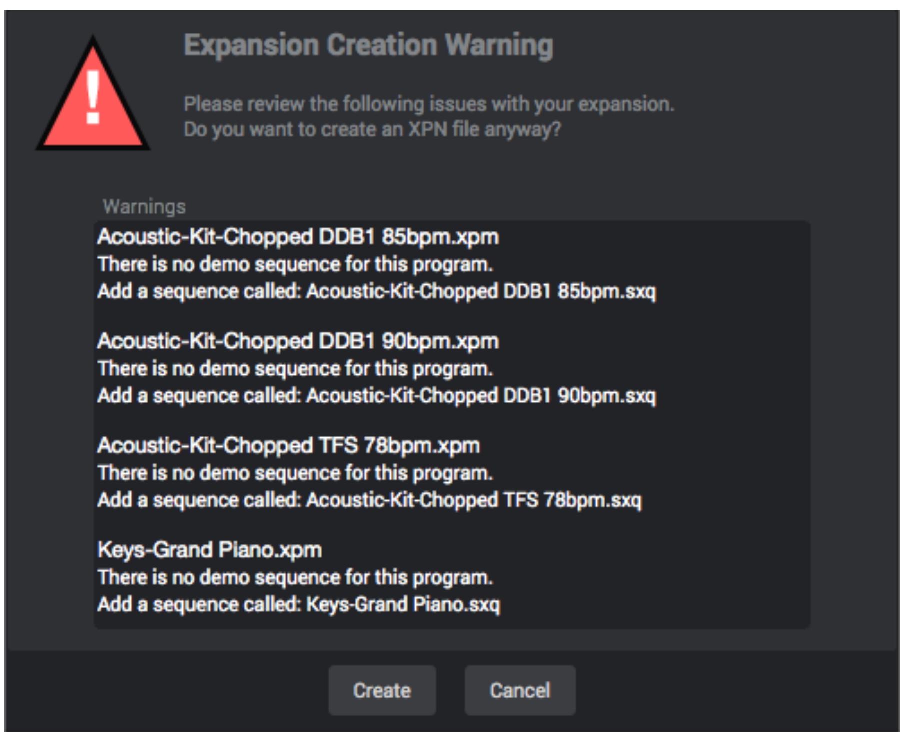
Finally for '**Content Directory**' navigate to and select the location of the '**MPC Bible Expansion Files**' folder.

The **description** field is used during installation to inform the end user what the expansion is all about, The '**identifier**' is a unique name used by the MPC Software to identify your pack and its contents on the system and needs to be in a 'reverse-DNS-style' format. If you have a domain name, enter it in reverse, e.g. **com.mpc-samples**; if you don't have a domain name, just enter **personal.yourname**. Then enter the name of the expansion, replacing any spaces with full stops. We'll look at the **Name Separator** a little later in the tutorial.

So far, everything should look like this:



Hit **Do It** and you'll get the following warning:



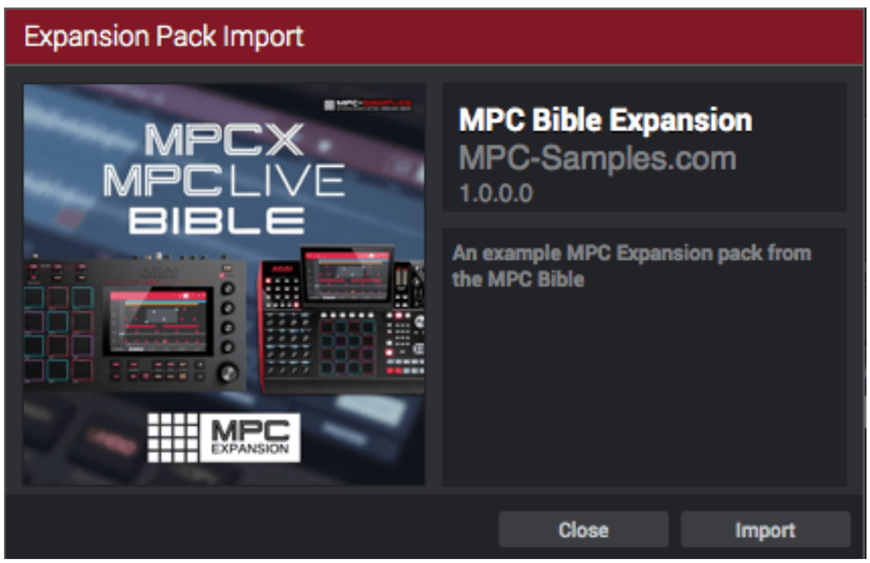
Here the MPC is offering you the opportunity to add a demo sequence for each program in your expansion pack. This is entirely optional so you can omit this step if you wish – however as we'll soon see, I have already created some demo sequences for three of the kits in our expansion pack, so I'll explain how to do this correctly later in this tutorial.

Hit **Create** and you'll be asked to give the file a name and select a location to save your XPN 'installer' file to. Call the XPN file '**MPC Bible Expansion Installer'** and choose any location on your computer (it doesn't matter where you save this).

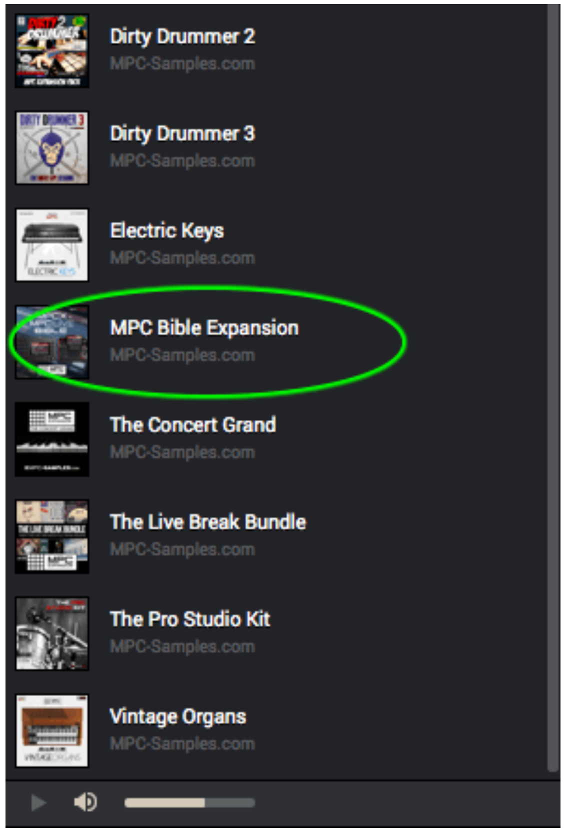
Hit **Save** and after a few seconds your **XPN** file will be created and placed in the selected location.

Installing an Expansion Pack From an XPN File

To install the expansion into the MPC Software, simply drag the XPN file directly into the GUI. You should see the following:



Hit **Import** to install the expansion. At this point the Expansion will appear in your Expansion Browser:

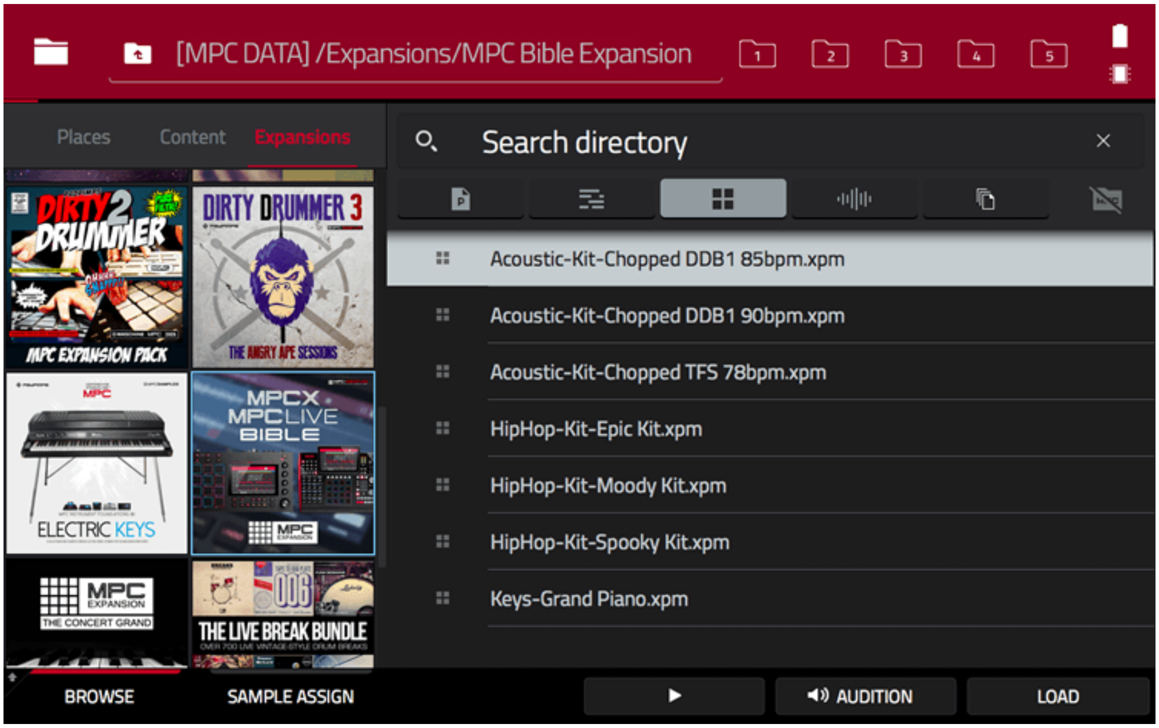


Click on the **MPC Bible Expansion** entry to see the contents:

The Hardware 'Expansion' Browser

The MPC hardware UI has very limited Expansion pack support, but it will recognise installed Expansion packs and organise them by thumbnail image, although browsing functionality is no different to the standard file browser.

To view the Expansions available, click on the '**Expansions**' tag on the left side of the screen:



Click on the Expansion you wish to view on the contents will appear on the right side of the screen. At this point the browsing experience is identical to browsing folders, so you'll need to use the file type filters and keyword search for further filtering.

In '**Controller**' mode, the Expansions listed in the browser are the ones already installed in your computer. In '**standalone**' mode you'll see the pre-installed factory expansions ('**Vault 2**' and '**Essential Instruments 2**') plus any third party expansions you have installed directly on your MPC disk.

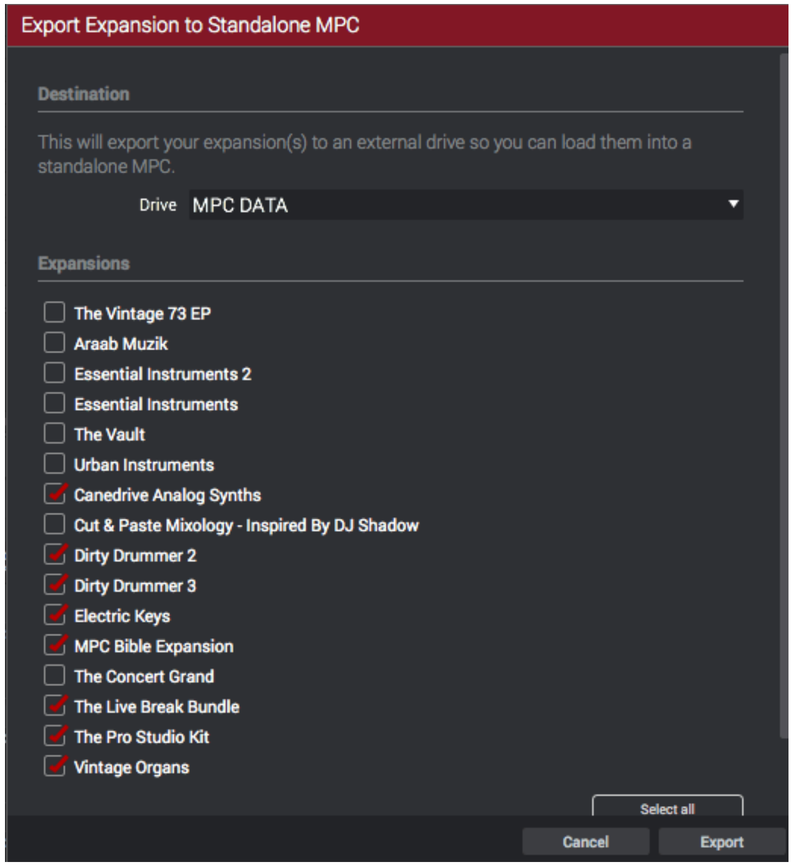
Installing Expansions in Standalone Mode

The pre-installed expansions in 'standalone mode' are stored in a folder called 'Expansions' on your 16GB internal drive but as we can't directly access this internal drive, we cannot copy additional expansions to this location. Instead we have to install additional expansions to our 'attached' disks.

Whenever you attached a disk to your MPC, be it a USB drive, SD card or internal SATA disk, the MPC immediately looks for a folder in the root location called 'Expansions'. If it finds any expansions inside this folder it will display these expansions alongside your factory expansions in the MPC browser 'expansions' tab.

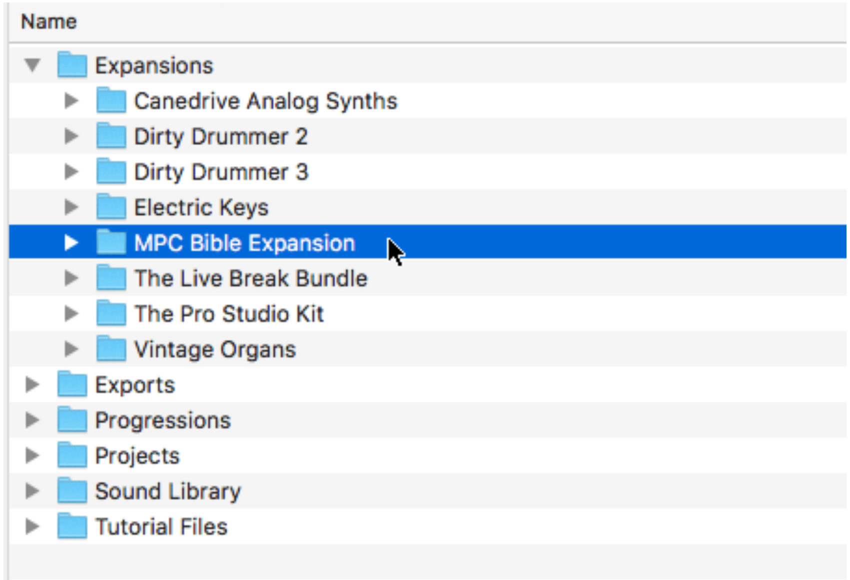
These easiest way to set up your expansions on an attached disk is to first install the expansion into the MPC Software. First connect your MPC disk to your computer; you could do this via controller mode (a requirement if transferring to an internal SATA disk) or you can attach a removable disk directly to your computer.

Then in the MPC Software go to '**File > Export > Expansion**'.



Select your MPC disk under '**Drive**' and then check all the expansions you wish to export to standalone mode. Hit **Export**.

After export your MPC disk will now contain a folder called '**Expansions**' and inside there will be a folder for each expansion you exported - these folders contain all the expansion content files:

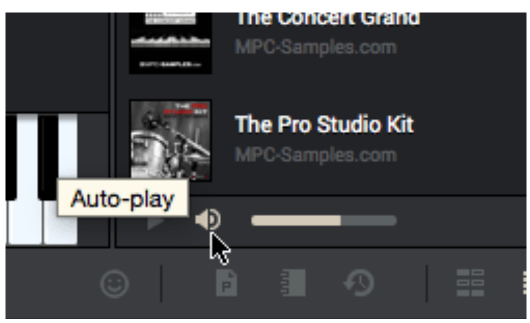


The next time you are in standalone mode, these expansions will appear in the browser - but only if your MPC disk is attached.

Creating Program Previews

When you click on a program in some expansions you'll hear an audio preview of that program; this is referred to as a 'Program Preview' and is a feature available in both software and hardware UI browsers.

Select the MPC Bible Expansion in the computer GUI expansion browser. If you don't have auto audition enabled, click on the speaker icon at the bottom of the Browser:



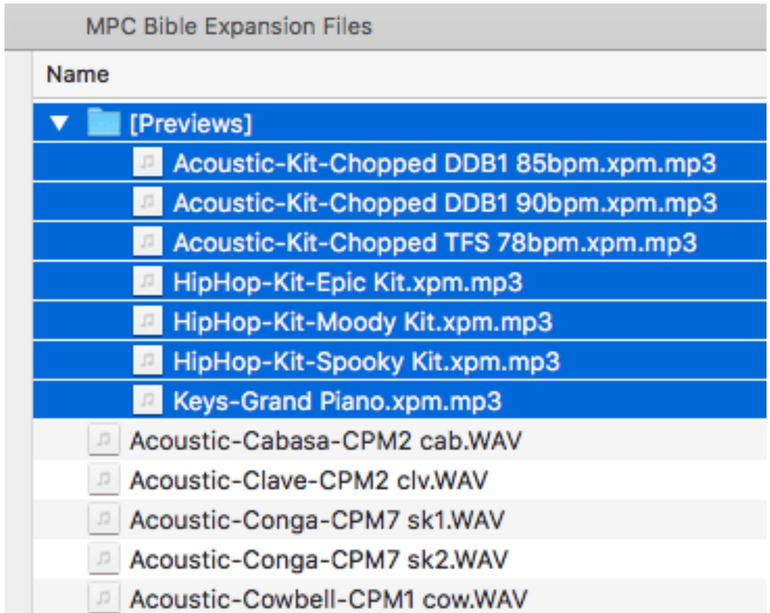
Now click on any of the programs in this expansion, either in the '**HipHop'** group or in the '**Acoustic'** group and you should now hear an audio demo play.

In the MPC Software this is referred to as a '**Program Preview'** and you can use this to quickly preview the sounds available in a program without the need to fully load it.

To create a program preview you'll first need to create a sequence that uses the sounds from a particular program – luckily we already have those three demo sequences I made previously, so I just used those to create the audio previews. With your demo sequence loaded and active, go to **File > Export > Audio Mixdown** and render your sequence into a stereo master (mp3, 192 bitrate will be fine, but you can use wav files if you prefer):

To use this mp3 file as a program preview, it's important to name the audio file after the full program name *including the xpm extension*; so for the Moody Kit demo we need to call the audio mixdown **HipHop-Kit-Moody Kit.xpm.mp3**

All your 'program preview' audio files must go into a special sub folder in the 'content' directory that you use to create your expansion pack. Navigate inside the **MPC Bible Expansion Files** folder. Here I had already created a sub folder called **[Previews]**:



Here you'll see the program previews I created for all the programs in the expansion; for the chopped break programs I just converted the actual original breaks themselves into mp3 format. When the Expansion Builder app installed the expansion pack it uses this '[Previews]' folder to set up the program previews throughout the pack. Program previews are however entirely optional – if you omit a preview, clicking on the program in the expansion browser will simply play nothing.

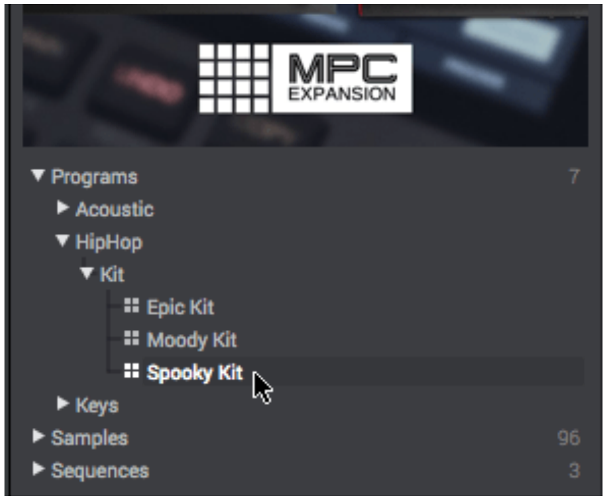
Please note: *The 'Program Previews' folder method also works when browsing folders with the standard file browser - just follow the same set up with a Previews' folder containing audio demos named after the program file name itself.*

Using Name Separators (MPC Software)

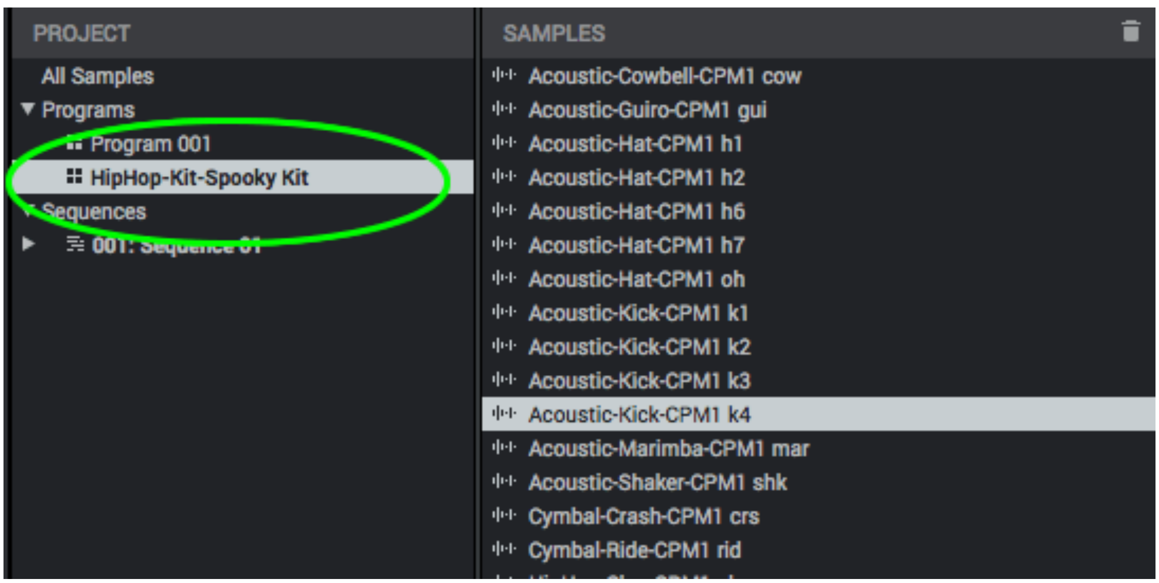
When setting up the expansion pack we entered a 'name separator' (in the shape of a hyphen). This name separator serves two purposes in the MPC Software. Firstly it is used to provide 'tags' in the Media Browser, and secondly it allows us to 'group' sounds and programs into categories within the Expansion Browser.

In the hardware browser there is no tagging or group support, however as tagging is incorporated into the file name of the program or sample, it can still be useful for search filtering.

Expand the '**Programs**' group to reveal three 'sub' groups; **Acoustic**, **HipHop** and **Keys**. Expand the **HipHop** group to reveal the three hip hop 'kits':



Load the **Spooky Kit** by double clicking it or by dragging it into your project. Now take a look at it in the **Project Panel**:



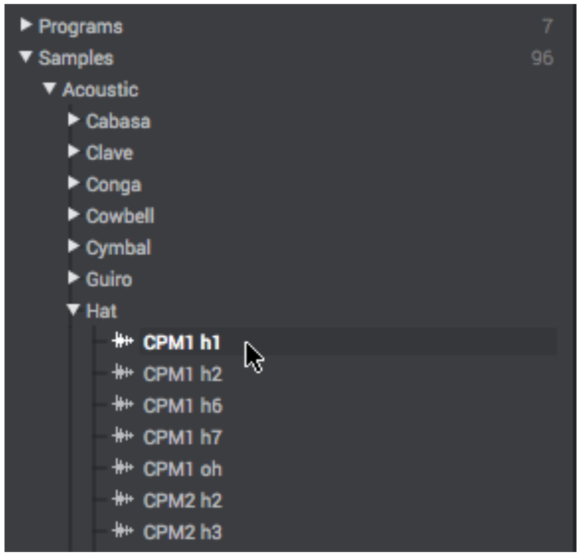
Notice how its name is not just 'Spooky Kit'? It's actually called '**HipHopKit-Spooky Kit**'.

If you set a name separator when creating your expansion pack, the MPC Software will look at the text before each instance of that name separator and will create a group from this. The structure is as follows:

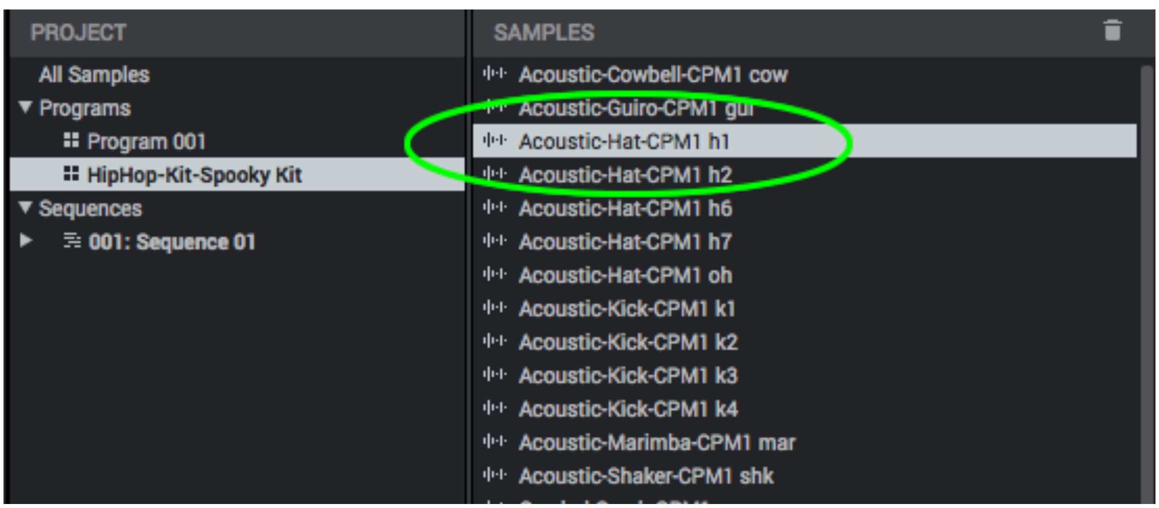
<primary group name><separator><secondary group name><separator><kit name>

So with our hyphen (-) acting as a separator, the MPC sees '**HipHop'** as the primary group name for this file and 'Kit' as the secondary group - these are then displayed as categories/groups within the Expansion Browser.

It's the same for the '**Samples**' category - expand '**Samples > Acoustic > Hat**':



Now load the **CPM1 h1** sample and view it in the Project panel:

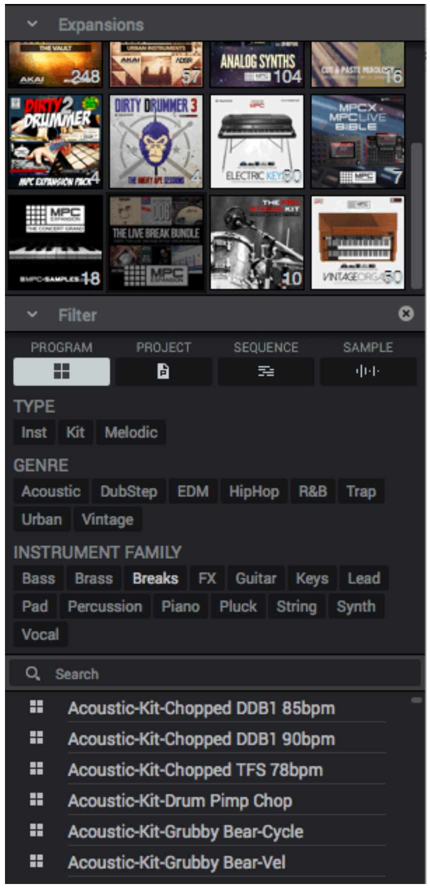


This hat sample is actually called **Acoustic-Hat-CPM1 h1,** but is displayed in the Expansion Browser as **CPM1 h1** within the groups '**Acoustic**' and '**Hat**', thus placing them all within the snare group.

You can place a file in as many groups as you wish, just use a hyphen to separate them all. Now, while you can use any group names you wish it's important to understand that these groups actually have a dual purpose as they are also used as ‘tags’ in the **Media Browser**.

Using Tags in the Media Browser

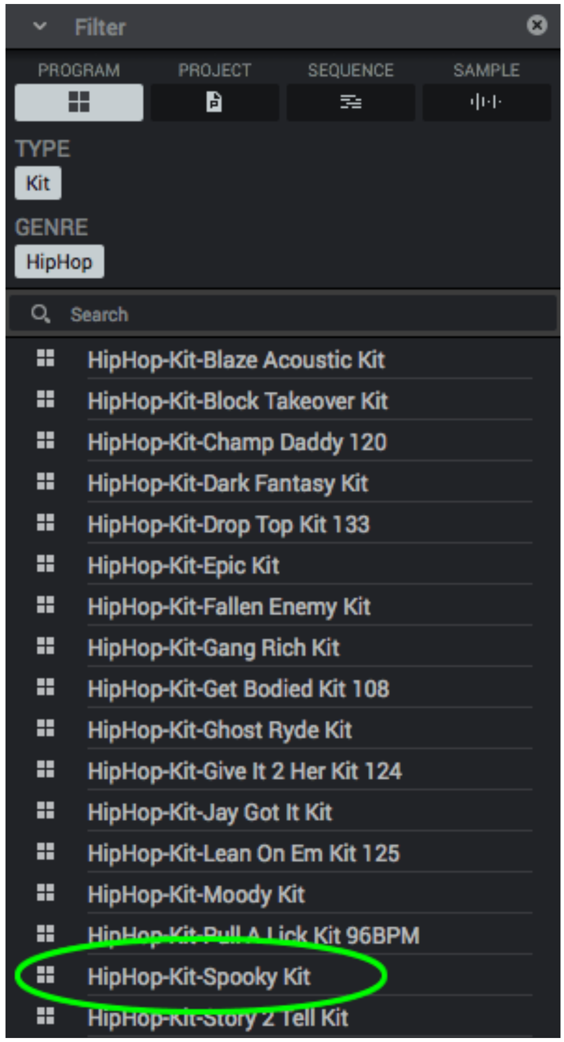
Hit **B** on your computer keyboard to open up the Media Browser.



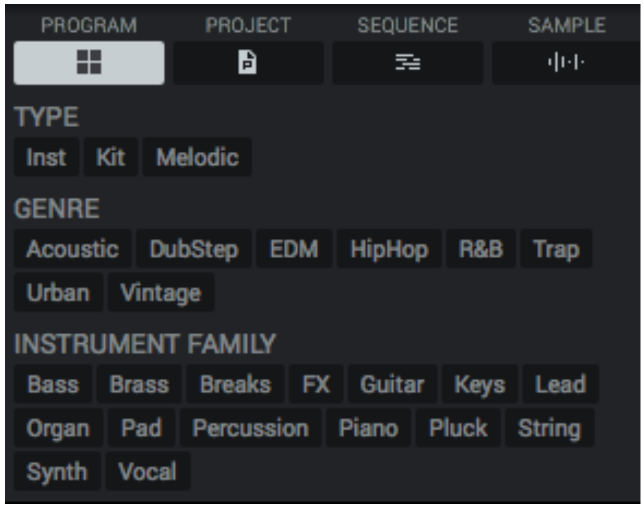
The Media Browser provides an interface to search for specific sounds or programs within your *entire* Expansion pack library - basically if the Expansion is installed on your system, be it an Akai or a third party expansion, it’s sounds, programs and sequences will be included in Media Library search.

A search box allows you to search for a specific phrase or keyword, but the ‘**Filter**’ section allows you to filter your Expansion collection by specific **tags**.

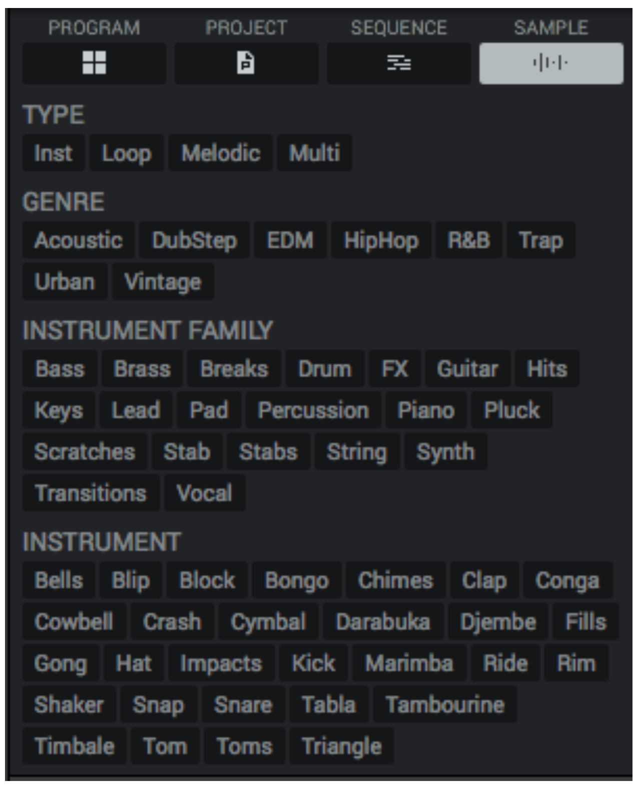
For example, make sure the ‘**PROGRAM**’ filter is selected, and then select **TYPE: Kit** and **GENRE: HipHop** and you’ll see a filtered list of all ‘Hip Hop kits’ in your Expansion library (or at leat, all the ones that have been tagged as HipHop kits) - here you’ll see **Spooky Kit**, **Epic Kit** and **Moody Kit** from the MPC Bible Expansion.



So you can see how the Expansion file 'groups' we set up are also being utilised as 'tags'. The tagging system employed by Akai actually requires you use very specific tag names, although currently there is no official list of accepted tags from Akai so the only way to know which ones to use is to examine the tags used in the official Akai expansions. For example, here's the 'Program' tags generated from expansions such as The Vault 2 and Essential Instruments:



And here's the 'Samples' tags:



So most common instrument types have a tag but **genre** tagging seems fairly limited. There may be other tags you can use but I've not found any yet.

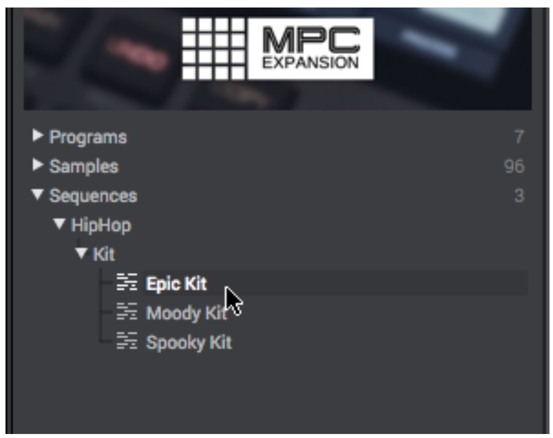
To be clear, you are of course free to use any tags in your naming structure, but only a specific set of tags can appear as selectable options in the 'Filter' section of the Media Browser. For example the file **Acoustic-Guiro-CPM1 gui** is an acoustic 'guiro' sample in the MPC Bible Expansion, but as you can see, the tag

Guiro does not appear in the recognised tag list, while other percussion instruments do (e.g. Timbale, Shaker etc). You can however still manually search for the term ‘Guiro’ using the search box.

Using tags is of course entirely optional, it’s a lot of work to have to rename all your files just to get them ‘tagged’ so I suspect most people simply won’t bother. Hopefully in the future Akai will introduce a tagging system like Maschine, in which tags can be batch applied to multiple files simultaneously.

Demo Sequences

Demo sequences are ready-made sequences that showcase a particular program. Head back to the Expansion Browser (X) and go to the ‘**Sequences’** category; expand the groups to reveal the **Epic Kit** sequence:



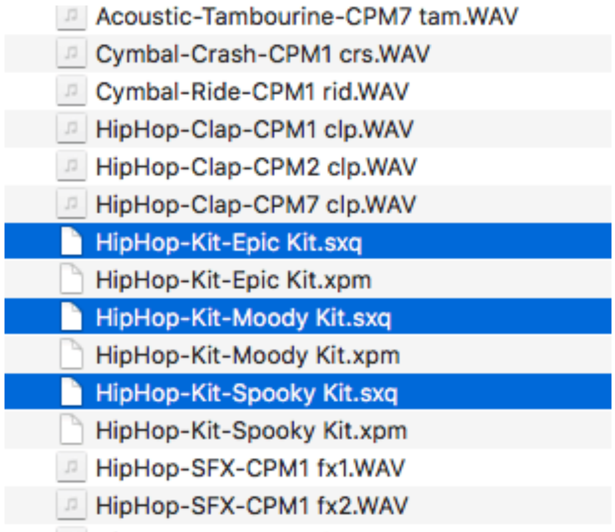
Double click (or drag and drop) this sequence to load it into your project. As the sequence loads you’ll see that the MPC also simultaneously loads the Epic Kit program.

You can hit **PLAY START** to immediately to hear the demo.

Creating a demo sequence is very simple. Start off with a blank sequence and load up the program you wish to create a demo with. Now…get sequencing and write a short demo with the kit – there's no limit on the length of the sequence, it can be 1 bar or 1000 bars, it's up to you. Once you've finished it go to **File > Save Current Sequence** and give the sequence the exact same name as the program it is demoing.

So, the Epic Kit program is called **HipHop-Kit-Epic Kit.xpm**; its demo sequence must therefore be called **HipHop-Kit-Epic Kit.sxq**. Save the demo sequence to your expansion pack 'content' folder. When the Expansion Builder creates the expansion XPN it will see there is a demo sequence for this kit and it will associate them together accordingly.

If you navigate to the original content directory we created the XPN from you can see my demo sequences were already there:



Remember the 'nag' screen we had when creating the XPN file? We got this because the Expansion Builder noticed there were no demo sequences for our chopped break programs – that's fine, demo sequences are optional and the MPC will create the expansion pack regardless.

You can of course include other sequences in your expansion pack – basically any other sequence file included in your 'content directly' will also appear under the '**Sequences'** category, but unless they are exactly named after a particular program, the MPC will just treat these like any normal sequence, so will load it purely as a sequence (it will not load any associated programs simultaneously).

Uninstalling/Removing Expansions

To remove an expansion from the standalone browser, go to the Places tab on the left side of the BROWSER screen and select your MPC disk. Enter the **Expansions** folder and single tap on the folder relating to the expansion you wish to remove. Hold down **SHIFT** and select **DELETE FILE > DELETE**.

In the MPC Software you can manage the installed expansion packs by going to **Tools > Expansion Manager**:

If you wish to *temporarily* remove any expansion, just uncheck its red check box – this leaves the expansion on your hard drive but makes it no longer visible from the expansion browser.

If you wish to *uninstall* an expansion pack, click on its entry and an 'uninstall' button will appear – however, this option is only available for third party 'XPN' installed packs.

Official Akai packs cannot currently be uninstalled using the expansion manager. If you really wish to uninstall an Akai expansion pack you'll need to manually delete the pack from in the following locations on your hard drive:

Mac: Library > Application Support > Akai > MPC
PC: C:\Program Files\Akai Pro\MPC\

Once inside you'll need to look at two different folders; '**Expansions**' contains the pack image and the pack configuration file (XML), while '**Content**' contains the actual sound content folders for each pack. To remove an official Akai pack you need to delete its specific 'content' folder *and* you need to delete it's XML file in the 'Expansions' folder.

After uninstalling the pack, restart the MPC Software.

Appendix D: Essential MPC Resources

MPC Forums: http://www.mpc-forums.com
The largest MPC community on the planet – run by yours truly!

MPC-Samples: http://www.mpc-samples.com
Continually providing fresh sounds, books & tutorials created specifically for MPCs

MPC-Tutor.com: http://www.mpc-tutor.com
Free tutorials news and reviews for the MPC world.

MPC Tutor on Facebook: http://facebook.com/mpctutor

MPC-Tutor on Twitter: http://twitter.com/mpctutor

MPCStuff.com: http://ww.mpcstuff.com
The one-stop resource completely dedicated to mods, upgrades and spares for all MPCs.

Akai Pro: http://www.akaipro.com
Official Akai web site

Appendix E: Book Release History

1.3.0

Release date: 4th May 2018
Summary: Free update to cover many of the new features in 2.1 and 2.2 firmware updates, with over 100 pages of new content and numerous tweaks to existing content throughout the book.

- **Chapter 001** - Minor text and image updates
- **Chapter 002** - Added sections about boot up/new project options. Project templates. Updates to browser screenshots to show new ‘expansions’ tab, minor text changes
- **Chapter 005** - Added info on relative vs absolute snap, added new copy/paste method, updates to some screenshots
- **Chapter 006** - Updated Q-LINK references
- **Chapter 008** - Updated QLINK references
- **Chapter 009** - Updated LIST EDIT section to reflect new INSERT method and ability to now edit MOD TYPE. Other minor changes.
- **Chapter 010** - Minor improvements to text
- **Chapter 011** - New sections on pitch quantise, random note generation and humanize. Added custom progression tutorial. Minor updates to plugin bounce tutorial.
- **Chapter 012** - Minor updates to text and screenshots
- **Chapter 013** - Screenshot updates
- **Chapter 015** - Rewrites to chopping workflow section
- **chapter 016** - Updates to grid screenshots
- **Chapter 017** -Added tutorials on drum humanisation and generating random drum events. Chapter renamed.
- **Chapter 018** - Minor text updates
- **Chapter 022** - Added ‘psuedo stereo’ example
- **Chapter 026** - Completely new chapter for plugin ‘cloning’
- **Appendix C** - New appendix covering expansions

1.2.1

Release date: 10th November 2017

- Optimised file sizes of EPUB and MOBI books.
- **Chapter 000** - Added link to my article that explains how to use the EPUB and MOBI books in mobile devices.

- **Chapter 005** - Updated the two final sections (PDF pages 96-102) to fix some mistakes in the screenshots and text.
- **Chapter 021** - Added workaround for saving individual samples from SAMPLE EDIT (PDF page 396)

1.2.0

Release date: 18th October 2017
Summary: Maintenance release for the 2.0.5 firmware update. Most of the changes are due to the new SAMPLES screen interface in PROGRAM EDIT, as well as the new LIST EDIT mode, GRID copy and updates to audio track editing features.

- **001 - The MPC Workflow**: Updated screenshots to reflect visual changes to MAIN
- **002 - Building a Simple Drum Kit**: Added missing BROWSER image. Various screenshot updates. Internal format option
- **003 - Recording Your First Beat**: Update to metronome options
- **005 - Quantising and Swing**: Added new section at end of chapter covering event & bar copying techniques
- **006 - Drum Layering**: Text and images updates for new PROGRAM EDIT > SAMPLES screens
- **007 - Drum Kit Essentials**: Fixed a couple of typos
- **009 - 16 Levels Bass Lines**: Updated for new LIST EDIT mode
- **010 - Keygroup Programs**: Screenshot updates for new SAMPLES screen
- **012 - First Look Audio Tracks**: Updated audio editing screen UI, fixed error in 'Multiple Audio Tracks.xpj' where sequence 2 accidentally loaded a 'reversed' audio track 2.
- **013 - The MPC Mixer:** MASTER channel screenshot updates
- **017 - Building Velocity Sensitive Drum Kits:** Screenshot updates to SAMPLES pages
- **018 - Dynamic Tempo Manipulation**: Updated to cover DRUM warping in standalone mode.
- **019 - Non Destructive Chopping**: Updated to reflect new SAMPLES screen PAD START/END UI, added section about applying WARP/STRETCH to problematic chops
- **020 - Progressive Drum Layering**: Updated for new SAMPLES screen and PAD START/END UI. Added 'flatten pads'.
- **021 - Sound Design Tricks**: Updated for new SAMPLES screen and PAD START/END UI.
- **022 - Advanced Keygroup Editing** : Updated for new SAMPLES screen.
- **023 - Create Your Own MPC Synth**: Updated for new SAMPLES screen
- **024 - Building Songs & Performances**: Added LIST EDIT examples for XYFX automation data
- **025 - Adding Vocals & Exporting Beats** Added LIST EDIT examples for Q-LINK automation data, updated Q-LINK references due to the way 2.0.5 has reversed some QLINK assignments. Updated audio track editing section to add guide on using the marquee tool as well as general improvements to the existing tutorial content.

1.1.0

Release date: 8th September 2017
Summary: Maintenance release to fix various typos within the book as well as inclusion of an indexed & linked 'table of contents' in the PDF version of the book, plus 'side bar' bookmarks for easy back and forth navigation (this will display in compatible PDF readers such as Preview and Adobe Reader).

1.0.0

Release date: 11th August 2017
Summary: Initial release.