

The EX5 Arpeggiator

Part 2 : How it actually works

In part 2 of the arpeggiator tutorials, we are actually going to discover "how the arpeggiator actually works". It can confuse many as its implementation is different from many synthesizers.

The first thing we need to understand is how normal music data is represented within the arp itself. YAMAHA call each note within the arp a **KEYCODE**.

=====

A **KEYCODE** is Yamaha's name for a note within the arpeggiator. The secret to it is remembering that it initially remembers the notes you play and converts them over to a number it can use to play should the initial keys be not pressed. That way, an arpeggio "feel" to what has been entered can be faithfully reproduced.

KEYCODE CHART

-Note- -Keycode-

C 1

C# 2

D 3

D# 4

E 5

F 6

F# 7

G 8

G# 9

A 10

A# 11

B 12

At this point of time. I want you to understand that these notes represent these KEYCODE's. We will move into what some typical EX5 arp data looks like.

=====

Here is what a portion of the data looks like for the EX5 ROM preset ARP known as "USR [Techno-A]". If you would like to look at this sort of data on the other pre-sets, you must follow the arpeggio copy procedure we did in the "EX5 ARP TUTORIAL 1". Then press [EDIT] from within the [Arp Edit] page.

TECHNO-A

001-01-000	1 +0	00-025 ()	80
001-01-120	1 +0	00-055 ()	80
001-01-240	1 +0	00-055 ()	80
001-01-360	2 +0	00-055 ()	80
001-02-000	3 +0	00-055 ()	80
001-02-120	2 +0	00-055 ()	80
001-02-240	3 +0	00-055 ()	80
001-02-360	2 +0	00-055 ()	80
001-03-000	1 +0	00-060 ()	80
001-03-120	3 +0	00-060 ()	80
001-03-120	4 +0	00-060 ()	80
001-03-240	2 +0	00-055 ()	80
001-03-360	1 +0	00-055 ()	80

Lets break a line of this data down for a better understanding. Take the first line of the arpeggio for instance.

1. "001-" : This is the number of the current **measure**.
2. "01-" : The number of the current **beat** within the measure.
3. "000" : The position of the current **clock** at the particular beat.
4. "1" : This is the **keycode**.
5. "+0" : Indicates the **octave**.
6. "00-025" : The note's **gate** time.

7. "()" : In between these brackets, is the note that closely represents the gate time.

8. "80" : This number is the **velocity** of the note.

You will also note that there are two events at "001-03-120". This is a chord.

=====

If you are a little confused still at this moment time, just bare with me. The EX5 arp has a few other components that affect the way its data is interpreted. This is where we now move to our machine and do the following:

Press the [VOICE] mode button. (This makes sure we are in the current mode).
Now we are going to dial up a particular sound so we hear what is going on.

Press the [F2] (which is [P2] on the LCD) button and dial up the voice "VCE P2-083(F03)". Otherwise known as "Ba:AnaChorus".

Press [EDIT].

Now we are going to COPY and existing ROM arpeggio to a "user" defined location and work on it. (This is as per the 1st tutorial if you're not sure what is going on.)

Press the [COM] button.

Press [ARP].

Now turn the arp "on" and dial up "Type = 051: USR[Init Arp].

Press [ARP-EDIT].

We will look at shortly this page. So, make a mental note that this is the area that we will be coming back to as this is where the "data" interpretation happens.

Press the [JOB] button.

Move the cursor to "1. Copy Arpeggio" and press the [ENTER] button.

Change the "Src Arp =" to "042: P[SeqS&H1]".

Now change the "Dst Arp =" to "051: U[Init Arp]".

Press [ENTER] then the [Yes/Inc] button. The arpeggio is now successfully copied from the existing ROM location to a "user" area.

Press the [EXIT] button twice.

Play the keyboard to make sure we have that funky/weird computer arp happening!

Now we have a VOICE programmed with an ARP we want to look at. SO we can continue with the tutorial.

=====

You should be in the page I told you to make a mental note of.

The first to take note of is the **TEMPO**.

When programming an arp, the default **TEMPO** setting is located here. If you change it to a number like "83" for instance, any VOICE's that you program with this arp will start with

83 when you turn it "on from the [ARP] page. To get an idea of what I mean, press [EXIT] now.

You will see that the **TEMPO** is set at 83. Change it here to 120.
Now press [ARP-EDIT].
The **TEMPO** is set at 83 in here.

This is made this way so you can indicate to people that you have programmed this arp this slow for a reason. Therefore, it will default in the main [ARP] page when programming a VOICE.

But heh! The arp's are there to pump up to 200bpm's. ☺

=====

Now lets look at the data within the arp itself and see the many ways that the EX5 can interpret it. Press [EDIT].

You should now be within the data for **TRACK 1**. If it is another track, then press the [COMMON] button on the far right side of the EX5 This will light up the LED. By the way, the 4 numbers on those buttons are the indicators for what current **TRACK** you are in.

The first thing you will notice is that on the first line there is a MIDI message. (BRIGHT). In addition, it's current value. (26).

001-01-120 074:Bright 26

Any MIDI control change data that is contained within the arp will be obeyed by the EX5. It will affect all **ELEMENTs** within a **VOICE**. It is something you need to consider when programming. A list of all MIDI control messages the EX5 will respond to is contained on page 41 of the EX5/EX5R DATA booklet that came with your machine.

The main reason we are here, is that I wanted you to identify with this type of data been contained within the arp. It is extremely useful, as we will see now.

Press [EXIT].

What we are now going to look at is the "Key=sort" function.

The **KEY** function instructs the EX5 on how it should use the data. The 3 modes **KEY** can be set to are:

1. **SORT** = Instructs how notes will be interpreted should more than one key be pressed. In this case, it instructs the arpeggio to play from the lowest pitch to the highest. (This function we will be coming back to.)
2. **THRU** = In this case, the EX5 will play keys as they are played on the keyboard. So its looking at which key will be pressed first. It can be a matter of milliseconds between keyboard strikes. For example, a chord.
3. **DRCT** = is short for DIRECT. This mode allows you to use only the MIDI control messages in the data. The arp doesn't play its pre-instructed sequence at all. Things like PAN, BRIGHTNESS, and HARMONIC content, along with all the other MIDI control messages it can play back. (Again, refer to page 41 of the EX5/EX5R DATA booklet.).

This function is particular useful. Since the EX5 doesn't have synchronized LFO's, you could emulate it on a filter by using these types of control messages. The EX5 existing arpeggios do not give a hint towards this type of flexibility. As they all are programmed to use the **SORT** function.

Lets look at the **DRCT** function since our data contains the MIDI control message 74 in it. (Brightness).

Change "Key=" to **DRCT**.

Now change the **TEMPO** to something like 83 so you can hear things better. Play a note on the keyboard, especially in the lower registers and hold it. You can hear the effects of the **BRIGHTNESS** on the filter of this sound. Note that the arp is playing, its just not triggering notes!

=====

One more thing I would like to explain on this page is the "Vel = seq ". This is instructing the EX5 arpeggio how you want velocity messages to be interpreted.

There are 2 modes for **VELOCITY**:

1. **SEQ** = This is short for SEQUENCE. When set to this, the arpeggio will play back the velocity data contained (pre-programmed) in the arpeggio. The velocity you strike on the keyboard itself is not used at all.
2. **THRU** = This allows us to use the velocity we actually play. It is handy if we have velocity ranges programmed within **ELEMENTs** that have different zones. This way, you could have a sequence playing a piano and a string on different keys depending on how hard we play the keyboard. The usual method has been to use **SEQ** as the mode and trigger from the different velocities from within the sequence itself. That way it is always a "controlled" series of velocity events. But by using **THRU**, you can control it yourself.

Change "Key =" to **SORT**. Then change "Vel =" to **THRU**.

Now play the keyboard with varying velocity strikes. Notice the difference in the way it plays back.

=====

With some of the "jigsaw" puzzle of the EX5 arpeggiator put together, we can now set out to explain the operation behind the **KEYCODEs**.

What we are going to do is look at some data as it appears in music notation, and then look at it as it appears in the EX5.

Here is the opening bar of Guns'n Roses hit song "Sweet Child O' Mine".

Now if this data were used in the EX5 arpeggiator, in a very raw state it would look like



this.

001-01-000	3 +0	00-216 ()	72
001-01-240	3 +1	00-216 ()	72
001-02-000	10 +0	00-216 ()	72
001-02-240	8 +0	00-216 ()	72
001-03-000	8 +1	00-216 ()	72
001-03-240	10 +0	00-216 ()	72
001-04-000	7 +1	00-216 ()	72
001-04-240	10 +0	00-216 ()	72

It so happens, that included in the ZIP file that contained this document, is a file called TUTOR_2.S1R. This file contains this arpeggio. So unzip it onto a floppy disk and put it in the disk drive.

Press [DISK].

Now press [LOAD] followed by [ARP].

Look for TUTOR_2.S1R and press [ENTER].

Now press the [YES/INC] button.

Press the [VOICE] mode button.

We will try this on a "guitar" voice, so press [P2] and dial up "P2-050(D02)". Which is "GT:EX Guitar".

Press [EDIT], then [COM], followed by [ARP].

Turn the arpeggio to "on". And change the "Type =" to "051:USR[Gun's 1]".

Play the keyboard if you like. You will notice that nothing happens. This is the basis for our entry into understanding **KEYCODEs**. Press [ARP-EDIT]. Now press [EDIT].

We are now in "Arp Edit" of **TRACK 1**.

If you press the "V" (the down arrow on the keypad), and keep pressing it all the way to the bottom of the track, you will hear the riff play.

Why didn't it play to begin with?

Ah hah! We now have got to the nitty-gritty of **KEYCODEs**.

=====

If you look above at the data (or on the LCD screen of your EX5), you will see a list of all the **KEYCODEs** used at each position of musical time. The **KEYCODEs** used in this instance are 3,3,10,8,8,10,7,10.

The EX5 uses numbers to associate what keys have been pressed. If you look at the musical notation of the first note you will it is a "D". Now if you compare it against the "KEYCODE CHART" at the start of this tutorial, you will see its **KEYCODE** is that of No. 3.

If you look at the second note of the notation, you will see it is also a "D", with the exception it is an octave up. Look at the EX5's representation of the data and you will see that the **KEYCODE** is still 3 and that it is followed by "+1". This indicates 1 octave up.

Why didn't the notes play?

Well, the problem with the EX5 arpeggiator is not only does it associate notes with numbers, it associates the amount of keys you have pressed with it also. This means we have to press 3 keys down to play just the first note!

If a "C" note was present in the notation (KEYCODE of 1), then the first note would have played.

Before we move along from here. Press down the B, C, and D keys together on the keyboard. You hear all the notes associated with the KEYCODE no. 3. Now while holding these 3 notes down, press down the A key below the B key. (You should be holding down 4 notes.) Notice that the tune moved down when we did this.

This is now due to the fact that "Key =" is set o **SORT**. (Pitches are played from lowest to highest).

So our data would look at A as KEYCODE 1, B as KEYCODE 2, C as KEYCODE 3, and D as KEYCODE 4. Since we do not have any **KEYCODEs** associated with the numbers 1,2 and 4. Nothing is played.

Remember this, our arp is not looking for the "D" at all now. Its looking for the third key pressed down on the keyboard.

Things are a mess now, as our data will wait for 7,8 and 10 keys to be pressed down before it will play any other notes in our sequence. What is the solution?

If you have read the "JOY of EX", you will find what it does with an arpeggio a little confusing.

Before moving on, remember these points:

1. Once you've entered your sequence correctly. Forget its relationship with musical notation.
2. You are going to have to "massage" the data to get it working in a similar way to your original sequence.

Ok. The first thing we have to do is get a note to play when we push our first key. To do this, we are going to have to edit some of the **KEYCODEs**.

Press [EDIT].

Change all the **KEYCODEs** with the number 3 to 1. (Use the arrow keys to navigate your way to the **KEYCODE**).

The reason I chose to change 3 to 1 is that they are the next lowest notes in the **KEYCODEs**. The fact that they were the first couple of notes in the sequence was also an added advantage.

Next **KEYCODE** we will change is that of 7 to no. 2. Since it is the next lowest of all the **KEYCODEs**.

Now change all the no.8 **KEYCODEs** to that of no.3.

To see how things are going, press [EXIT]. Now play a three fingered chord using the notes D, F#, and G on the keyboard. Play different inversions of the chord to see what happens to the sound. Remember that the **KEY** is set to **Sort**. So we are playing from lowest to highest pitches. So introducing lower notes will change the way the arpeggio behaves. As the EX5 will then associate **KEYCODE** no.1 with the lowest note and then count up.

Press [EDIT] and change the remaining **KEYCODE** from 10 to 4. The arpeggio is now complete. Press [EXIT] and press a chord with the notes of D, F#, G and A. You will hear a representation of the original data that was entered in to begin with.

Obviously, if the original pattern had been more complex. We would have had more **KEYNOTES** to begin with. In addition, it would have required the use of more fingering on the keyboard to get what was originally entered.

But this way, you can capture the feel of another song and use it in your arpeggio.

=====

Before we conclude on this arpeggio, we will introduce something that is standard among many of the existing EX5 arpeggio's in ROM.

You should be still in "Arp Play" mode. This is where the "Key =" and "Vel =" part of the arpeggio is. You will notice that on the bottom of the LCD is a button called [MODE].

This is where you can further affect the way the notes play on the keyboard.

Press the [MODE] button.

There are 17 selections in total. You can make an arpeggio scatter itself over as many as 4 octaves through using one of the **MODEs**.

As standard, it is set originally to "00:non search". This means the data plays back as normal.

Most arpeggios in the EX5 presets are set to "01:Search Low". This will play **KEYCODE 1** (the first key you push down) in the position of other **KEYCODEs**. It never replaces those keycodes, it just plays a note at every keycode that isn't being triggered. The idea behind it is to allow a note to be played all the time should only 1 key be pressed down.

For each of the 4 tracks in an arpeggio, you can choose a **MODE** for each of them.

Try changing the **MODE** of track 1 of our arpeggio to "01:Search Low". Now press down one key.

You will now hear the affect it will have on the data and the way it is played back.

Press down the D, F#, A and G keys together, and the original programmed sequence is still there.

=====

As per the first tutorial, we are going to now play around with the **PLAY EFFECTs** on the EX5. If you have forgotten how to work it, and what it actually does, then you will have to refer back to it.

Move the cursor over to the "FxThru" block under 1 and make it *hollow*.
Press [PFX].

Change it to "Typ= 019:16AcidJazz". Play and listen to the different **PLAY EFFECTs**.

You can do plenty on your EX arpeggiator. Now there is not any reason why you cannot get in and starting making your own. Don't forget to share them with the rest of the EX community. ☺

=====

Special thanks to Chris "The SynthMeister" Steller from YAMAHA Australia.

This concludes Part 2.
Copyright 2000 : Brian Cowell

12th November, 2000