

# **FunkTracker**

**FREWARE**

## **TABLE OF CONTENTS**

**A - The Editor**

**B - Command System**

**Specification**

## **Section A**

## **The Editor**

The FunkTracker editor has two edit modes of operation. The first is a *Sample Editor*. This section enables you to load and delete samples as well as change the attributes of those loaded samples in a given song. The second is the *Pattern Editor*, which allows to you edit patterns and maintain the Sequence table.

In order to use these samples, we must first load them. This can be achieved with the sample editor. When the tracker is first ran, it will by default be active. Here we can load or first sample. To load one, press the SPACE BAR. The tracker will then display a directory with a sticky bar. This will allow you to navigate through your harddisk and load a *Windows WAVE* (\*.WAV) sample file (or a raw sound file - \*.SND). This distribution provides you with some of these files. Load the first one you see by highlighting it with the arrow keys and pressing enter.

Once loaded, you can play it to determine if you like it. To play it, you can use the keyboard keys.

z	C--
s	C#
x	D--
d	D#
c	E--
v	F--
g	F#
b	G--
h	G#
n	A--
j	A#
m	B--

Alternatively, you can use a MIDI keyboard, when activated.

A Funktracker song can hold up to 64 of these samples that can be up to 4 Gigs in size. If you decided that you didn't want that particular sample, you can overwrite it by replacing it. Just move the highlighter over the sample you wish to replace and press the SPACE BAR again to load the alternative sample. If you wish to delete a sample, then press the DELETE key. Note that the editor fills the blank space by shifting the samples up. If you wish to insert a black entry then press the INSERT key.

As you may have noticed, samples have a number of attributes....You will see the fields in the sample editor. They are as follows:

Name	-A 19 char sample name field
Start	-Loop start. If FFFFFFFF, then sample doesn't loop.

Length	-Length of Sample
VL	-Volume of sample (256 point)
BL	-Balance of sample (256 point, 0 = left)
P	-Port type of Sample. 0 = Logarithmic porting, 1 = linear porting.
S	-Sample offset Parameter (Used by Command N and a few others).
V	-Vibrato WaveForm type
V	-Tremelo WaveForm Type
R	-Note Retrigger Limit (number of beats to do)
A	-Arpeggio Speed of Command L (arpeggio)

We can change these attributes by first moving the highlighting over one a desired field using the arrow keys and pressing the [ key. I'll only discuss some of these attributes, as most of them are used by the command effects. The rest of them will be discussed in Section D.

The Sample Name is 19 characters dedicated to assigning a name to a sample. To the computer, this field is irrelevant. All it does is contain a short sample description. It can contain whatever you like. The Sample Length field is an informative field that displays the size of the loaded sample. As the size of fixed, you can't change this attribute. It's for information purposes only. If the length of a sample is zero, then that means the sample entry is unoccupied. The Start field determines if a sample loops or not. If this field is set to a FFFFFFFFh (in hexadecimal) value then the sample when played will stop when it has reached the end of play. If this field not equal to this number, then it will *loop* back to an offset equal to this number, and continue playing it. The VL field determines the volume of that sample. It has a value range of between 0 to ffh (hexadecimal - 0 to 255 in decimal). The BL field determines the balance of the sample. The balance is a gradient between 0 and ffh. All the way left is 0, and all the way right is ffh. 80h is half way, which means that the sample will play in both left and right speakers at equal volume. This attribute is only reserved for soundcards that are capable of advanced digital sound processing (DSP cards) like GUS and AWE32. The cards that are excluded from this are the SB,. SBPRO and the PAS family. When using these soundcards, the balance is determined by the channel the sample is played in. This complies with the 669 balance standard of left-left-right-left-right-left-right-left-right.

Once we have loaded the samples we want (or just a few of them, whatever..), we can go to the pattern editor and start actually keying in a tune. To display the pattern editor, hit the F2 key. Once there, you will see an empty pattern. You will also see the sequence table. To move around this pattern, you can use the arrow keys. Using page keys in place of the up and down arrow keys does exactly the place thing accept a lot faster. Now we are really to enter in our first note. Hit any one of the keyboard note keys. Once pressed, you should hear one of the samples being played. It will play the currently highlighted sample in the sample editor. Alternately, this sample is displays down the

bottom of the screen in the pattern editor, and there you have it! :). You continue to do this until you have a tune. If you want to change the sample you are playing, then use the o and p keys. If you want to change the octave of the notes you are entered, then use the / and \* keys. You can also, shorten a given pattern by setting the end of pattern marker to a desired slot row. You can change the sample number of an existing slot by hitting the [ key. Note that if you enter a sample number into a Null slot, it will change that slot into a Sample Only Slot. Pressing the ' key will set a "RLO" (Reload Sample attr) slot (Note, for information on 'slot types', refer to next section).

You can also copy sections of the pattern and paste that copied area just like you can in a word processor. To do this, you must first select the area that you wish to copy. Go to the leftmost/topmost slot of the block you want to copy and press Alt-b. Then move the highlight over to the rightmost/bottommost slot and press Alt-e. Once the area has been selected, press Alt-c. Now, that area can now be pasted in any pattern in any song module. To paste over any area, move the highlight wherever you wish to have the area pasted and press Alt-o. Once you have edited a pattern, you may like to do and edit another. To change patterns, you can use the + and - keys.

Now that you have edited some patterns, you will be required to enter them into the sequence list. To do this, use the HOME and END keys to move up and down the table. Once you have a chosen position, press the 9 key. It will enter the currently displayed pattern into the pattern list. If you want to backspace over an entry in the sequence table, press the 0 key. To set the repeat location, use the Alt-s key to set a given sequence position to be "looped back to" when the tracker has reached the end of the sequence table.

To play an entire song, press either F5 or Shift-F5. The two keys give different types of displays. To play a song from the currently highlighted sequence position (ie the yellow lighted number in the sequence list), press either F6 or Shift-F6. To stop a currently playing song, hit the Escape key. You can silent the individual channels by pressing the keys 1 thru to 8.

### **Where do I go now?**

The best thing to do now, is go play a few modules and see how they work. Observing a real life module in operation is better than any of my notes. The above was just a rough guide. There you will see how the commands work, and how you use the patterns. I would now go and use funktracker to make a couple of songs. Once you have become familiar with the editor and the general format, come back and read the next section on Commands, as it may be a bit to complicated for you now.

## **Section C** **Specification**

## **Command** \_\_\_\_\_ **System**

This Section is devoted to describing the 31 or so Funktracker commands at your disposal. Funktracker been implemented with a new concept when playing effects. That is, commands have been split up into groups. These different groups have the ability to play independently without ceasing or interfering with each other.

### **Slot Types**

Before we go on, just a small discussion on slot types. In Funktracker a slot is defined as a single unit of data that the tracker reads instance per instance for each channel when reading a pattern. A pattern is made up of a grid of them. Anyway, there are 4 types of them:

#### **FULL SLOT:**

This type of slot defines a sample, a note and a optional command. This is a slot that you will most frequently use the play a note. When the tracker decodes this type of slot, it loads the sample defined by the number at the specified note. You can set one by playing a note on the keyboard.

#### **NULL SLOT:**

This slot is a "nothing" slot. Although Void, it's still a slot :). Only defines a optional command. To set one, press the space bar.

#### **SAMPLE ONLY SLOT:**

This command defines only a sample number and an optional command. Here the tracker loads the sample and starts playing it without resetting any of the channel registers like "Frequency", "note" or "Volume". This attributes would normally be changed when using a Full Slot. You can set one by moving to a NULL SLOT, and setting a sample number by pressing the [ key and typing a number.

#### **RLO SLOT:**

This command loads sample attributes into a given channels registers. The sample data loaded is defined by the number. Optional commands can be used with this slot.

## **Command Mechanics**

The command handling of FunkTracker is like that of the 669 standard. Continual commands are not processed for the one slot like ProTracker. Funktracker will react upon a command and continue acting upon that command until it reacts another command (in the same system), or it reaches a full slot. Lets now compare two typical channels of the ProTracker and FunkTracker format. See below:

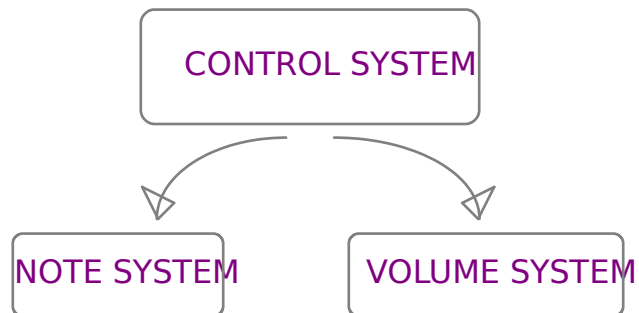
Protracker	FunkTracker
00 C#101ARP45	00 00 C#1  45
01       ARP00	01       ---
02       ARP00	02       ---
03       ARP00	03       ---
04       ARP00	04       ---
05       ARP00	05       ---
06       --- --	06       o0A
07       --- --	07       ---
08       TRM4F	08       k4F
09       TRM4F	09       ---
0A       TRM4F	0A       ---
0B       TRM4F	0B       ---

The two channels (with their respective contents) will do exactly the same thing. As the Protracker muso will tell you, the tracker will read slot 00 of which will play sample 01 at C#1 with a command of Arpeggio-45. Now if you wished to continue this command over more than one slot, do would have to set those slots to that command. In Funktracker, it will continue to apply that command over slots 01 to 05 without any repeated instructions to do so. At Slot 06, The Protracker will stop the Arpeggio command as there is no command instruction in slots 06 and 07. FunkTracker will also stop the Arpeggio command. It will do this when it reads the command "o0A", which is a command that halts any current command of the note system.

So in summary, Protracker (and most other Formats), process commands on *instance*, where is FunkTracker processes them *marginally*. The method of processing commands has been inherited from the 669 format. This method allows for concurrent processing of different types of commands, which is this commands most powerful feature. Below is an example of this dual command piping:

00	00 C#1 I45
01	k4F
02	--
03	--
04	--
05	--
06	--
07	--
08	--
09	--
0A	--
0B	--

The above example will play sample 00 as an arpeggio, and then both as an arpeggio and a volume tremolo in slot 02. This is possible, as this commands are split into 3 Possible Systems. Each of these systems handle commands in there own way and have there own separate register values. When a system is executing a command, it won't effect the execution of any other system. A diagram of these systems and their communication dependencies are below:



The Control system acts as a front end for both the Note System and Volume System. It reacts to the patterns as if it was a ProTracker player (*ieinstance* by *instance*). If It comes across a command that doesn't belong to it, it will pass that command over to the appropriate system. The Note System controls/governs a set of commands that manipulate the frequency of a channel. The Volume System controls/governs a set of commands that manipulate the volume of a channel. The Note System cannot cease, start or interfere with the Volume System, nor can the Volume System interfere with the Note System. Finally, a Control System command cannot cease a Note System or Volume System command, but can in some cases interfere with these systems operations (which is in most cases desirable).

### **The Commands "A" thru "N"**

Now you are aware of how FunkTracker handles its commands, this section will describe each of them.

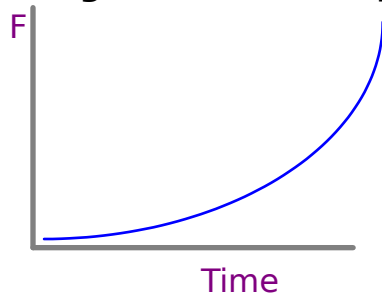


**Command A:- "Port frequency up" System)**

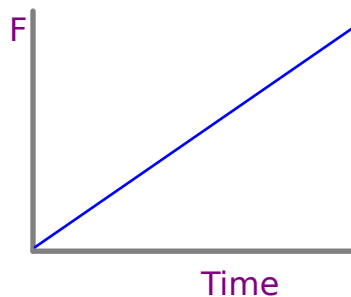
**(Note**

The *Port Up* command effect will slide (increase) the frequency of a given channel. All types of note porting have two types of modes. It can either port in a straight linear fashioned, or a curved (bent) logarithmic fashioned. This is set by a later referenced command. Below is a visual diagrams on how the frequency of a channel of effected when using this command.

**\* Logarithmic Port Up**



**\* Linear Port Up**



For those of you that are experienced, Protracker (and many other Trackers) implement logarithmic porting. COMPOSD1/UNIS669 (669 format) on the other hand implements linear porting. FunkTracker has both!.

The Command has a dual parameter. The first nibble of the parameter is the "rate" in which the frequency will slide up. The Bigger the number the greater the rate of slide. The second nibble is the "speed" parameter. This is the speed in which it will do the slide.

**Related Commands:**

B, C,00A,00C,00E,00F,04,05

**Command B:- "Port frequency down"  
System)**

**(Note**

This command is exactly the same as Command A - Port Up, except that it slides the frequency of a channel downwards.

**Related Commands:**

A,C,00A,00C,00E,00F,04,05

**Command C:-"Port To Note"**  
**System)**

**(Note**

This command can only be used in "Full Slots" (if you use it in any other types of slots, then it will have no effect). It ports a the current note of a given channel to the frequency of the given full slot. This can be a port up or a port down. It depends on the source frequency of the channel and the destination frequency of the note. The parameters and mode of these command are exactly the Same as Command A.

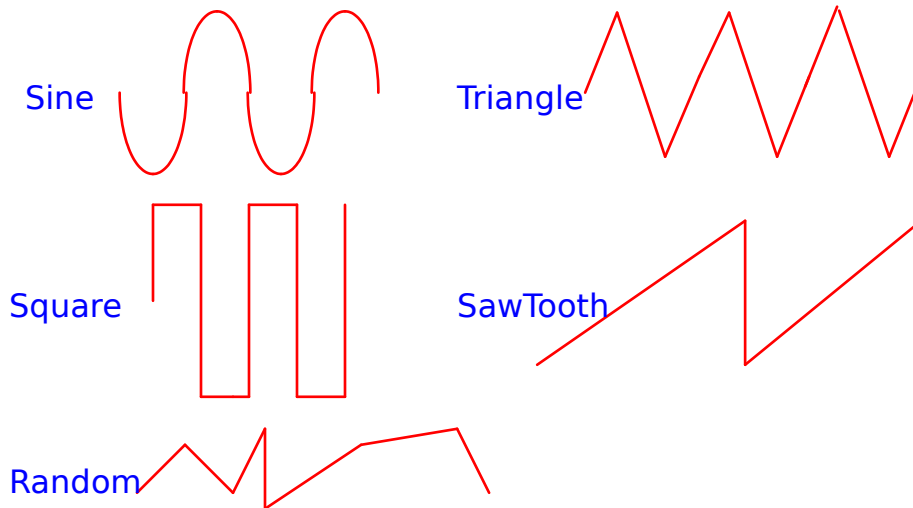
**Related Commands:**

A,B,00A,00C,00E,00F,04,05

**Command D:- "Vibrato"  
System)**

**(Note**

This Command vibrates the frequency of a given channel. The first nibble of this command parameter sets the speed in which it vibrates the frequency. the second nibble specifies the amplitude of the vibrato. Vibrato by default vibrates the frequency in a sine wave fashion. However, it can also vibrate the frequency using other wave functions. These are below.



Changing the wavetable involves a command the is referenced later in this section.

**Related Commands:**

E,F,000,001,002,003,004,00A,00C

**Command E:- "Vibrato Fanin System)**

**(Note**

This command is an extension to the Vibrato (Command D) command, and should be used after this command. its job is to slow down the vibrato over a period of time. It will slide the speed from the last set vibrato speed to the maximum possible speed for Vibrato (that being 0). The parameter for this command (like the last) is dual. The First nibble specifies the speed of the slide, the second nibble specifies the amplitude of the vibrato. Note, if you don't use this command after a vibrato, then a start vibrato speed of Fh is assumed. See below:

00	00 C#1 d45
01	---
02	---
03	---
04	---
05	---
06	e0A
07	---
08	---
09	---
0A	---
0B	---

00	00 C#1 a45
01	---
02	---
03	---
04	---
05	---
06	e0A
07	---
08	---
09	---
0A	---
0B	---

Start Speed is "4" at the last  
Command is a Vibrato command  
A Vibrato speed of "4"

The last command isn't a  
Vibrato, therefore, a start  
speed of "F" is assumed.

As with Vibrato, this command can have different waveforms as well.

Related Commands:

D,F,000,001,002,003,004,00A,00C

**Command F:- "Vibrato Fanout"  
System)**

**(Note**

This command has exactly the same characteristics as Command E "Vibrato Fanin", and is intended to be used with the Command D "Vibrato" command. The acceptance is that it slides the speed of the vibrato up, hence speeding it up. It will slide the speed from the last set vibrato speed to the minimum possible speed for Vibrato (that being F). If the last command isn't a vibrato, then the start speed of the vibrato is "0", which is the fastest vibrato speed.

**Related Commands:**

D,E,000,001,002,003,004,00A,00C

**Command G:- "Volume Slide Up"**

**(Volume**

**System)**

This command will slide the volume of the channel upwards. This command is a dual parameter command. The First nibble specifies the speed of the slide, the second specifies the rate. The command will stop when the volume FFh has been reached.

**Related Commands:**

H,I,J,K,N,00B,00C,01,06,07,08,09,0A,0E

**Command H:- "Volume Slide Down"**

**(Volume System)**

Exactly the same as Command G, except that it slides the volume of a channel downwards. The command will stop when it reaches zero.

**Related Commands:**

G,I,J,K,N,O0B,O0C,O1,O6,O7,O8,O9,OA,OE



**Command I:- "Volume porta"**

**(Volume**

**System)**

This command is exactly the same as the previous two commands. It is only to be used in Full slots and Sample Only Slots (if you use it in "Null" slots, then it will have no effect). It will slide the volume from the current volume to the sample volume specified by the slot. Again, the parameter of this command is the same as the last two commands.

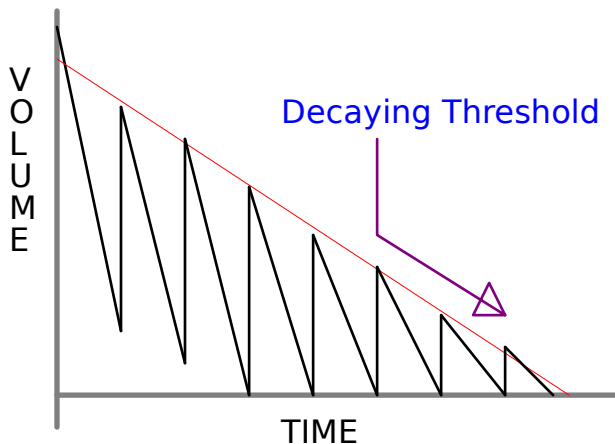
**Related Commands:**

G,H,J,K,N,O0B,O0C,O1,O6,O7,O8,O9,OA,OE

**Command J:- "Decaying Reverb"**

**(Volume System)**

This is a new command that I think you will use a fair bit. The idea has been adopted from the MIDI format. This format simulates an echo effect. Once activated, it will slide the volume according to a given rate, over a period of time. The Period will be set according to the first nibble of the parameter. When the time period has expired, the volume will be set to the original volume "minus 1". This minusing of the pecks, is why it is called a *decaying* reverb,. Here is a graphical presentation of a decaying reverb:



The second nibble is the rate in which the volume slides.

This command is quite a little beauty. It produces a very realistic echo. Just for names sack, it took me 3 different implementations to get the echo exactly how I wanted it to sound. This decaying ramp echo sounds the best :).

**Related Commands:**

G,H,I,K,N,O0B,O0C,O1,O6,O7,O8,O9,OA,OE

**Command K:- "Tremolo"  
System)**

**Volume**

This command has exactly the same characteristics as Command D (Vibrato), except that his command applies to vibrating the volume. The first nibble of this commands parameter Sets the speed in which it vibrates the frequency. the second nibble specifies the amplitude of the vibrato. For detailed discussion of the wavetable generator, see Command D.

**Related Commands:**

D,G,H,I,J,N,O0B,O0C,O1,O6,O7,O8,O9,OA,OE

## **Command L:- "Arpeggio" System)**

**(Note**

For those of you that are experienced with the Amiga formats, this command is exactly the same as the ProTracker arpeggio command (although, no code was ported from Protracker, so it may be a little different). This Command simulates a chorus effect, by switching the frequency of a channel between three different notes. The first note is the set note for the channel. The other two notes are actually specified by the parameter of this command. The parameter is a dual parameter, and each nibble is dedicated to a note value. This value is an offset value format the actual note of the channel (NB/ the actual note of the channel is the note set by the last full slot).

SLOT VALUE OF THIS ARPEGGIO: 00 C-1 I8F



Referring to the diagram above, arpeggio, will play 3 notes, one after the other. The first note will be the value of the slot, which is "C-1" (see 'base Note" on diagram). The second note that will be played, is specified by the first nibble of the parameter. This has 8 as it's value. Therefore it is 8 bars from the base note which makes it "G-1" (See NoteOffset1). The Second parameter has the value of "F" (which is 15 in decimal). Therefore, it is a note 15 places from the base note. This note is "D-2" (See NoteOffset2).

Unlike Protracker Arpeggio, this arpeggio can have it's speed varied. However this isn't specified here. One of the sample attributes is dedicated to defining the arpeggio speed of the sample, and here is were you will set it. However, you can override this setting with Command 03 "Set Arpeggio Speed". The default speed for arpeggio is "3".

### **Related Commands:**

03

**Command M:- "Override Sample Offset"**  
**System)**

**Control**

When a sample is played, it starts from 0 and ends at where ever This command allows you to set the start offset of a sample in a "Full" or "Sample Only" slot (if you use it in "Null" slots, then it will have no effect). The Parameter is a factor that determines the offset. This factor is calculated in the following formula to derive the actual offset:

$$\text{Offset} = \text{Parameter} * 2^{\text{SAMPLE OFFSET SHIFTER}}$$

The SAMPLE OFFSET SHIFTER value is stored as a sample attribute which can be changed in the sample editor. It has a default value of 8 (which is the equal to the constant value of the Protracker shifter).

**Related Commands:**

OB,OC

**Command N:- "Set Volume"**  
**System)**

**(Volume**

This command sets the volume of a given channel. This command is controlled by the volume system, and therefore halts any currently running volume system commands. Volume value can range from 0 to FF (or 0 to 255 in decimal).

**Related Commands:**

G,H,I,J,K,00B,00C,01,06,07,08,09,0A,0E

## The Command "O"

### Command 00:- "Misc Control" System)

(Control

This is a miscellaneous control command that allows you to set the various modes for the systems as well as one or two other things.

**000**        **vibrato sine**  
**001**        **vibrato triangle**  
**002**        **vibrato square**  
**003**        **vibrato sawtooth**  
**004**        **vibrato random**

The above 5 boolean control settings set the mode of the Vibrato command and all of it's extensions (See Commands D, E and F)

**005**        **tremolo sine**  
**006**        **tremolo triangle**  
**007**        **tremolo square**  
**008**        **tremolo sawtooth**  
**009**        **tremolo random**

The Vibrato controls, these 5 set the mode of the tremolo command (Command K).

**00a**        **halt note system**  
**00b**        **halt volume system**  
**00c**        **halt Note and Volume system**

Because the Note System and Volume System handle commands in a marginal fashion, there would be no actual way of halting a command without playing a new sample. These commands will cease any currently active command that is running. Command "00A" will cease only Note system commands, Command "00B" ceases any currently active Volume System Commands, and "00C" ceases any currently active command.

**00d**        **invert funkctrl**

This command is an override and it effects the way in which the sample is actually played back. You can only use it in a "Full" or "Sample Only" slot. It will invert the mode of looping. Therefore, if a given sample in a slot was looping, then this command will cause it to play as a nonlooping sample. If a given sample in a slot was non-looping, then this command will cause it to play as a looping sample (0 start loop assumed).

**00e**        **algorithmic porting**  
**00f**        **linear porting**

This two commands allow you to change the mode of port "on the fly" after a Full or Sample Only slot has been played. You could do a linear port (as set by the sample being played) and suddenly change that port to a logarithmic port. Or, you may wish to do a series of complex ports after a sample has been played using a combination of linear and logarithmic ports.

*Related Commands:*

Just about every single other command :)



**Command 01:- "Volume Cut"**  
**System)**

**(Volume**

This command is like the Protracker Volume Cut. It sets the volume of a channel to zero after a given period of time. The time period will be determined by the parameter.

**Related Commands:**

G,H,I,J,K,N,O0B,O0C,O6,O7,O8,O9,OA,OE

**Command O2:- "Real Frequency Adjust"  
System)**

**Control**

This command is like Command D of the 669 format. It sets a real frequency adjustment of the overall channel. When playing two samples at the same time, it gives the sample.....well, you'll have to hear it yourself, as the effect is too hard to explain here :). It's caused when you play two samples that are slightly of different frequencies. Unlike the 669 format, Once you set an adjustment, it will always stay, until you ever set it to zero or change it.

**Command O3:- "Set Arpeggio Speed"**

**(Control System)**

This command is an override that sets the speed of the arpeggio command (Command L). Each time a sample is played, the arpeggio speed from the sample attributes is loaded. This command is for the purpose of changing it "on the fly" after a sample has been played.

**Related Commands:**

L

**Command 04:- "Fine Port Up"**

**(Control**

**System)**

This command is a component of Command A "Port Up". It slides the frequency of a channel upwards. The only difference is that it will only do so once per slot. In other words, it's instanced to one single slot, and will increase the frequency according to its parameter value and then finish. Like Command A, it also has a Linear mode and Logarithmic mode. *(NB/ Command A uses this function to perform the actual porting by continually calling this function)*. Because this command only effects the frequency once, it isn't included in the Note System, and therefore is controlled by the immediate System - The Control System.

**Related Commands:**

A,B, C,00A,00C,00E,00F,05

**Command 05:- "Fine Port Down"  
System)**

**Control**

This command is exactly the same as Command "04" except that it fine ports the frequency of a channel downwards.

**Related Commands:**

A,B, C,00A,00C,00E,00F,04

**Command O6:- "Fine Volume Slide Up"  
System)**

**Control**

This command is a component of Command G "Volume Slide Up". It slides the volume level of a channel upwards. The only difference is that it will only do it once per slot. In other words, it's instanced to one single slot, and will increase the frequency according to its parameter value and then finish. *(NB/ Command G uses this function to perform the actual volume sliding by continually calling this function)*. Because this command only effects the volume once, it isn't included in the Volume System, and therefore is controlled by the immediate System - The Control System.

**Related Commands:**

G,H,I,J,K,N,O0B,O0C,O1,O7,O8,O9,OA,OE

**Command 07:- "Fine Volume Slide Down"**

**(Control System)**

This command is exactly the same as Command "06" except that it fine slides the volume of a channel downwards.

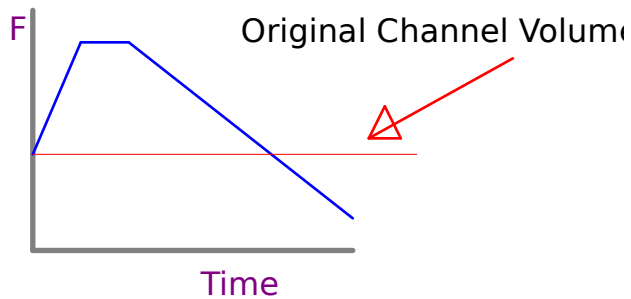
**Related Commands:**

G,H,I,J,K,N,O0B,O0C,O1,O6,O8,O9,OA,OE

**Command 08:- "Volume Crest"**

**(Volume**

**System)**  
The next two commands (as far as I know), are unique to this format. They simulate the attack/decay/sustain/release volume levels of a typical synthesised sample. Although primitive, are quite handy to use. Starting at the current volume level of a channel, it will be to slide it upwards, when it has reached the maximum possible volume level (FF), it will wait there for a period of time and then being to slide the volume back down to the original volume level for that channel. The slide up phase is always faster than the slide down phase, and the speed of the crest is determined by the parameter of this command.



**Related Commands:**

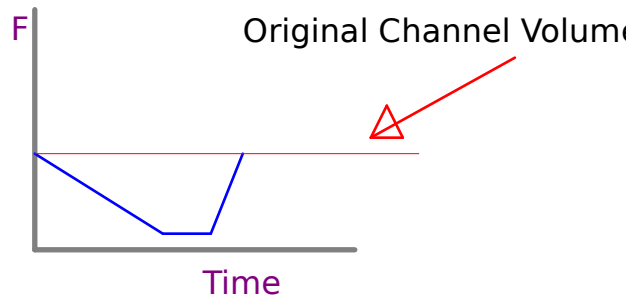
G,H,I,J,K,N,O0B,O0C,O1,O6,O7,O9,OA,OE



**Command 09:- "Volume Trough"**  
**System)**

**(Volume**

This command is exactly the opposite.



**Related Commands:**

G,H,I,J,K,N,O0B,O0C,O1,O6,O7,O8,OA,OE

**Command OA:- "Set Master Volume"**

**(Control**

**System)**

Master effects all volumes. Its effects the overall output from the soundcard as to fade the music up or down. It is used to fade in the volume at the start of songs, and fade the volume out at the finish of songs. The parameter is the adjuster value. The larger the value, the louder the volume of the overall output. The default master volume level is F.

**Related Commands:**

G,H,I,J,K,N,O0B,O0C,O1,O6,O7,O8,O9,OE

**Command OB:- "Expand Loop"**

**(Control**

**System)**

This command is used to effect the loop back of a currently looping sample of a channel. This command decrements the start loop of the sample in real time. The decrement is calculated in exactly the same manner as Command M:

$$\text{Start} = \text{Start} - \text{Parameter} * 2^{\text{SAMPLE OFFSET SHIFTER}}$$

This command is useful for special effects.

Related Commands:

M,OC

**Command OC:- "Collapse Loop"  
System)**

**Control**

This command is exactly like the previous command, except that it increments the sample start pointer. The increment is determined by the following formula:

$$\text{Start} = \text{Start} + \text{Parameter} * 2^{\text{SAMPLE OFFSET SHIFTER}}$$

This command is useful for special effects.

**Related Commands:**

M,OC

**Command OD:- "Note Retrig"**

**(Control**

**System)**

This command is the only continuous command of the Control System. It's a command created for use with percussion type samples. Over a duration of a slot, it will rapidly keep on replaying the sample, over and over again. The parameter is the speed in which it does this. The number of times it will replay the sample is determined by a setting of the sample attributes. The default for any sample is 4 times (but you can set this to whatever you like in the sample editor). This command (unlike other continuous commands like volume slide etc), won't play for more than the duration of a slot. You have to handle it like a Protracker command.

**Command OE:- "Set Channel Balance"**  
**(Control System)**

This command sets the panned position of a slot. As the command parameter is only a nibble, and the actual balance parameter is a byte, the command multiplies this parameter by 16 to get the actual pan value. Therefore, a Parameter of "F" is actually "F0", a parameter of "7" is actually "70" and so on.

**Related Commands:**

G,H,I,J,K,N,00B,00C,01,06,07,08,09,0A

**Command OF:- "Set Tempo"**  
**System)**

**(Control**

This command sets the speed in which the tracker "tracks" (ie the playing speed). The parameter specifies the speed. The lower the number, the faster the track speed, the higher the number, the lower the track speed (NB/ The tracker has a default speed of 4).

ooo0ooo  
END OF DOCUMENT