

## **Symphonie - Help**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> Symphonie - Help		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 22, 2024	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Symphonie - Help</b>	<b>1</b>
1.1	Symphonie Help Manual . . . . .	1
1.2	Where to get new modules for Symphonie . . . . .	1
1.3	program history . . . . .	2
1.4	(C)opy Right . . . . .	5
1.5	Schema eines zukünftigen Symphonies . . . . .	5
1.6	Warum, Wo erhältlich ... . . . .	6
1.7	über Int. Money Order . . . . .	9
1.8	über Cash . . . . .	9
1.9	über Symphonie Innereien . . . . .	9
1.10	über Soundqualität . . . . .	10
1.11	Einschränkungen der Pre Release . . . . .	12
1.12	GUI Hilfe : Inhaltsverzeichnis . . . . .	12
1.13	Menu Verzeichnis . . . . .	13
1.14	Info über : Menupunkt ? . . . . .	13
1.15	Info über : Menupunkt System . . . . .	13
1.16	Info über : Menupunkt File . . . . .	14
1.17	Info über : Menupunkt Edit . . . . .	15
1.18	Info über : Menupunkt Move . . . . .	16
1.19	Info über : Menupunkt Sample . . . . .	17
1.20	Info über : Menupunkt Prefs . . . . .	17
1.21	Blockoperationen . . . . .	20
1.22	System Control Info . . . . .	20
1.23	Song - Help . . . . .	21
1.24	Sequence - Help . . . . .	22
1.25	Position - Help . . . . .	22
1.26	Pattern - Help . . . . .	23
1.27	Track . . . . .	23
1.28	Instrument . . . . .	23
1.29	Waveform . . . . .	25

---

---

1.30 NoteEd . . . . .	25
1.31 Pattern Editor Feld . . . . .	30
1.32 Tastaturbelegung . . . . .	32
1.33 Sample . . . . .	33
1.34 GUI: Gruppe - Begriffserklärung . . . . .	33
1.35 Virtuelle Instrumente - Begriffserklärung . . . . .	34
1.36 MIX (kreieren eines virtuellen Instrumentes) . . . . .	35
1.37 MIX (kreieren eines virtuellen Instrumentes) . . . . .	35
1.38 CLONE (kreieren eines virtuellen Instrumentes) . . . . .	35
1.39 RVS (Parameter eines Samples) . . . . .	35
1.40 Downsample (Parameter eines Samples) . . . . .	35
1.41 FILT (Parameter eines Samples) . . . . .	36

---

## Chapter 1

# Symphonie - Help

### 1.1 Symphonie Help Manual

Willkommen zu Symphonie Junior v2.0 (2.03)

Einführung    Was ist Symphonie ?

Qualität    über Amiga Soundqualität

Internes    Innereien von Symphonie

Helpmenu    Hilfe ! Gadgets, Menus, Tastatur ...

History    Entwicklungsgeschichte von Symphonie...

-----

Module    woher bekomme ich Demo Stücke

Zukunft    von Symphonie ...

Copyright    Rechte

### 1.2 Where to get new modules for Symphonie

Sound Modules können bezogen werden bei:  
(2 US\$ / Module)

Marco Ege  
Rigiblickweg 5  
CH - 6048 Horw  
Switzerland

Tel. +41 4147 2162

Bisher erschienen von M. Ege

Songname    Modus    Länge (\*) (unpacked)

---

"Breathless"	8 Channel	300 KB
"Capella Remix #1"	8 Channel	1.8 MB
"Capella Remix #2"	8 Channel	1.6 MB
"Deep Vale"	8 Channel	700 KB
"Endless Ways"	16 Channel	570 KB
"Liquid Time"	8 Channel	370 KB
"Logical Dreams"	8 Channel	300 KB
"Rainbow Run"	8 Channel	500 KB
"Real Fantasy"	16 Channel	700 KB
"Revenge of Chapell"	16 Channel	700 KB
"Signing Off"	8 Channel	900 KB
"Virtual Harmonies"	16 Channel	910 KB
"Wild Canyon"	8 Channel	330 KB
"Devil Hunting"	16 Channel	100 KB
"No Way Out"	16 Channel	133 KB
"Party"	16 Channel	120 KB

(\*) Bem: Symphonie Modules erreichen Packraten zw 30% - 50% (xpkshrink/lha)

### 1.3 program history

Symphonie Junior V2.0

-----  
16BIT PREPROCESSOR - 16 BIT MIXER - 9BIT AUDIOOUT

V0.01 June 93 First Test Code  
(Mixing 32 Channels at 28Khz on A1200 !!!)

V1.01 ViRT Samples  
V1.04 IPOL Filter  
Position Transpose  
V1.053 Module/Extract Samples  
V1.054 Macro  
V1.055 Autoloop  
V1.057 Screenmode requester  
Oversampling

V1.058 less knacks  
nomore sync problems  
\*.WAV autoconvert algorithm

V1.059 All virtuel operations (Filter, Downsample ...)  
also on non virtuel samples

V1.060 Max Frequenz is now 65 KHz

V1.x Sample Preprocessor is fully 16Bit  
Sample tune range expanded to -/+24 halftones (4 octaves)  
Antiknack algorithm rewritten (now does a 32 sample vfade)  
NEW FX: "FromAdd", "FromSet", "SetSpeed"  
Multiwindow handler implemented

V1.3a channel mixing algorithm rewritten and expanded to 16BIT  
channel mixing is now ready for 16Bit soundcards (toccata is 1st)  
channel mixing now has FASTMEM BUFFERING implemented

-> users with FASTMEM get an extra bonus of 10% to 100% depending  
on their "CPU to FASTMEM" speed  
symphonie can now run at up to 250 kHz mixingrate !!!  
(if you are able to handle high monitor freqs)

GUI handler expanded to handle more object classes  
GUI is font sensitive

DSP rewritten and expanded for sequenced control

NEW FX: "PitchAdd", "VolumeAdd", "SetDSP", "Tremolo"  
"Vibrato"

Note Editor rewritten, is now object aware

#### V1.4 Sequence structure fully implemented

"SampVib" FX implemented, can do a vibrato like effect in  
the sense of sample (just rotates the samplebegin pointer)

Screenmode.requester: minimal screensize set to 320x200  
for users without vga/multisync monitors  
Oversampling 9Bit now is error free (no knacks)  
now using audio.device to allocate channels (no audio conflicts)  
VBR problem fixed (now using setvector to allocate interrupt)  
OS problems removed

symphonie now runs perfectly on CPUs:

68020  
68030  
68040 may need to disable 040COPYBACK using c:cpu  
68060 may need to disable 040COPYBACK using c:cpu

DSP chorus implemented for testing purpose (very buggy and noisy)  
DSP ready to have DELAY implemented

DSP system ready to handle multiple effects:  
eg. ECHO and CHORUS and DELAY in parallel !!!

STEREO CONTROL implemented:  
user can now expand the stereo base  
user can now also set a pitch offset between left and right channel

BLOCK commands fully implemented and bug-fixed

20 to 30 error messages implemented to inform user about  
what went wrong

#### V1.5 DSP Delay implemented (Echo & Delay result in complex Hall )

NEW: Symphonie can now perform Echo, Delay and ↔  
Chorus in parallel

Scope 1.0 implemented (for visuel sound analysis )

Prepacker implemented

-> extern compressors are now 100%-1000% faster ↔  
and may perform

---

better ratios

NEW: Prefs/Realtime/Force Update

NEW: Prefs/Realtime/Scope

NEW: DSP Control GUI (inside System Window)

NEW: Sample boost extended from max 100% to max 200%

V2.0 beta Assist implemented (Online Information)

Prefs save/load

Move Menu implemented

Prepacker extended for better prepacking

"PitchSlideTo" FX implemented,

"ShiftEmphasis" FX implemented,

"ReTrig" FX implemented,

V2.01 beta FIXED: Spectrum , SamplePosPtr

FIXED: Probs with large Patterns

NEW: Undo can now undo Undo (-)

NEW: Bpm expanded to Max 600Bpm

NEW: Noise Limiter Implemented (Prefs/)

NEW: Menu:Flag Autowindow to Front (if window gets activated)

NEW: Sample Volume above 100% now correctly implemented. Non destructive ←  
compressor

FIXED: Note Delete/Insert Fixed

NEW: Assist reports a "End of Sequence/Song"

NEW: Surround Left/Right expanded to 9Bit (50% less noise)

FIXED: Prefs:Force Update (no clock running ...)

BUG REMOVED: (Guru at End of Song) detected and Fixed

NEW: a lot of Info/Error Msgs redirected to use Assist as Output  
instead of using nasty OK-requests

NEW: Symphonie prepared(!) for 16Bit Samples

NEW: NoteEd now displays values as they act.

(e.g. Old: VibratoDepth 128 ==> NEW: ←  
VibratoDepth 50.0 %)

V2.02 NEW: Symphonie can now load Maestro 16Bit Mono/Stereo Samples

NEW: Symphonie can now load Wave(PC) 16Bit Mono/Stereo Samples

V2.03 NEW: DSP LP Filter for DSP Echo/CrossEcho

V2.04 NEW: System Buffer gets automaximized if the Buffer number is  
set too high

NEW: NEW DSP FX: CrossEcho2, CenterEcho (beta)

NEW: Symphonie can now load MAUD(IFF) Samples (8,16,Mono,Stereo)

First Symphonie Pro (100% 16 Bit Program)

Symphonie Pro is about 30%-40% slower than Symphonie Jr

Symphonie Pro needs double the memory of Symphonie Jr (16 Bit Samples)

V2.1 NEW: DSP FX Hall, CrossHall, CrossDelay

Delay LPFilter/Hall LPFilter

14 Bit Audio Out (Symphonie Pro only)



CPU Boost algorithm implemented (Symphonie Pro only)  
-> Symphonie Pro is 20% - 100% faster(!)

V2.1b          FIXED:    SAMPLEBOOST>100% didn't work in Symphonie Pro  
   (2.1 Plays 15Bit Sound, 2.1b Plays 16Bit ←  
   Sound)

```
FIXED: Knack in LPFilter
FIXED: Wrong "About Text" in Symphonie Pro
FIXED: Nasty Colorflash
```

HQ Mode implemented (more exact sound calculation, 50% slower)

## 1.4 (C)opy Right

Symphonie (C) by Patrick Meng - all rights reserved

Symphonie Demo (C) by Patrick Meng 1994  
Symphonie Demo is FREeware. No further limitations.

## 1.5 Schema eines zukünftigen Symphonies

Fernziel von Symphonie ist eine Integration in das Amiga OS  
ähnlich dem Grafik Subsystems des Amiga OS (graphics.library  
monitor - drivers, ...)

Das gesamte Sound Subsystem, besteht aus mehreren Programmteilen, die gleichzeitig im multitasking Betrieb arbeiten. Die Kommunikation erfolgt über spezielle Hochgeschwindigkeits Datenpfade.

Aufbau:

```

-----
Amiga OS x.0
-----

||
||
\ /

-----
Symphonie.library      Symphonie Processor      Symphonie Audio Drivers ↔
Hardware
    ---\
    ---\
    ---\

```

```
generiert 8/16/32/64Bit ---/ (DSP, Hall, Mixer, ---/ konvertiert aus den ↵
    Daten des      ---/ 16 Bit Soundcard
Daten ,              Oversampler...)          "Symphonie Processors" ↵
    Daten,          DAT
-----
Hardware           File (AudioRendering)
                                ----- ↵
                                "versteht" ↵
                                ----- ↵
                                ↵
/\
-----
||
||
\//
-----
Symphonie Synth Emulator
Emuliert Synthese Typen
-----
```

## 1.6 Warum, Wo erhältlich ...

Symphonie ist ein Produkt mit neuartigen Audio Routinen, die es ermöglichen sollen weit höherwertige Musik zu komponieren, als es bisher auf dem Amiga möglich war. (Auch in Games, Demos und Musiksoftware ...) Der Hauptvorteil von Symphonie ist jedoch, dass es alle vorhandene Audio Hardware unterstützen kann (Bsp: 16Bit Soundkarten, DSPs, 16Bit über Parallelport, etc).

Alle mit Symphonie geschriebenen Musikmodule werden von Symphonie automatisch an die jeweils vorhandene (Audio-) Hardware optimal angepasst (Rauschen, Quantisierungsrauschen, Samplingrate, CPU Speed).

Symphonie arbeitet 100% "non destruktiv"

Alle Samplemanipulationen wirken sich nicht auf das Originalmaterial aus !

Samples werden in Modulen so gespeichert, wie sie der Komponist auf seiner HD hat (unkonvertiert, ohne Manipulationen)

Die obigen Systemeigenschaften wirken sich folgendermassen aus:

Symphonie Module sind 100% hardwareunabhängig

Symphonie ermöglicht DDD Produktionen (voll digitale Produktionen ohne Verluste auf dem Signalweg ) ←

```
Beispiel: Lesen der Samples mit CD-ROM (Input)
          Verarbeitung mit Symphonie xy (non destruktiv)
          Exportieren in "High Quality" nach DAT (Audio Mode: ToFile)
          ("non realtime" Audio Rendering, 16Bit, 48Khz)
```

ev. Postediting mit HD Recording Software (destruktiv )

Symphonie Module können auf beliebige Hardwareplattformen skaliert werden.

Software Updates könnten die Klangqualität von schon existierenden Modulen "beliebig" steigern (falls entsprechende Hardware vorhanden ist)

Wenn ein Soundmodul mit 32Bit Samples mit 8Bit Symphonie (Jr) komponiert wird, so tönt es auf einem 32Bit Symphonie so, als ob das Modul mit 32Bit komponiert worden wäre !

Symphonie Hardwarevoraussetzungen:

CPU 680EC20 oder besser  
2MB Memory

Symphonie Softwarevoraussetzungen:

OS 3.0 oder höher  
regtools.library (im libs: Verzeichnis)

Empfohlen wird zudem:

Harddisk  
reichlich Fastmem

Symphonie gibt es in 3 Versionen:

-----  
Symphonie Junior : 80 US\$ (Release Datum: out now !)

- 
- 8...256 Soundchannels
  - 16 Bit Sample Preprocessor
  - 16 Bit Mixer
  - Audio OUT : 9 Bit Stereo, 3kHz - ca 72kHz

-----  
Symphonie Pro : (Release Datum: 1Q95)

- 
- 8...256 Soundchannels
  - 16 Bit Sample Preprocessor
  - 16 Bit Mixer
  - max 256x Oversampling
  - Audio OUT : 9Bit Stereo Amiga  
14Bit Stereo Amiga
-

16Bit Stereo Soundcard  
16Bit Digital Link (Rendering)

- Arexx interface
- Audio Rendering (for HDR, CD Production, DAT, Postmixing, Easy Synchronisation ↔ ...)

Modes : 8/16 Bit  
Mono/Stereo/L/R  
Normal/Intel Format  
Antialias On/Off

Format : Maestro (Samplitude)  
MAUD (16 Bit iff)  
8SVX (8 Bit iff)  
Wave (PC)  
Raw

Multichannel possibility (Mute Samples/Tracks/Dsp)

- Antialias algorithm for Audio Renderer
- Splitter algorithm for Audio Renderer (to calc multiple files instead of a ↔ single file)

-----  
Master Symphonie :           ()  
-----

- Midi
- Studio Quality
- Hall Designer
- Audio Rendering
- ...

Zukünftige Entwicklungsschwerpunkte (je nach Erfolg)

Neue Instrumenttypen ("Child", "Morph", "Multi" und "Complex")  
Neue Synthesetypen (FFT Resynthesis, FM, Multi)  
Echter Filter (Resonance, Freq, Tiefe)  
(Midi)

Die Zahlung erfolgt per INT. MONEYORDER od CASH  
Änderungen vorbehalten ...

-----  
Available at:

Realtime Software

---

Patrick Meng  
Rosenfeldweg 4  
6048 Horw  
Switzerland

+41 41486948 (Info, Development ...)

## 1.7 über Int. Money Order

Angaben:

BIC (Bankencode) : LUKB CH 2260A  
KONTO NR           01-08-303019-00 HORW  
NAME               PATRICK MENG

(USA:    via Citibank NY)

## 1.8 über Cash

Adresse:

Patrick Meng  
Rosenfeldweg 4  
CH-6048 Horw  
Switzerland

## 1.9 über Symphonie Innereien

Symphonie wurde geschrieben in 020 Assembler.

Intern arbeitet das Programm objektorientiert und ist quasi intelligent. Dies macht sich hauptsächlich in der Kürze des Programmes bemerkbar.

Die zentralen Routinen höchstoptimiert.

Die Restlichen Programmteile werden je nach Bedarf optimiert und befinden sich meist in der ersten oder zweiten Generation.

Symphonie System Daten:

Volumenauflösung       : 16Bit  
Frequenzauflösung      : 32Bit

Sample Preprocessor     : 16Bit  
ViRT Sample Processor  : 16Bit

---

Max 256x Oversampling

2 Phase Oversampling  
Software DSP

INPUT: (Samples)

Sampleauflösung : 8-64 Bit  
Samplingrate : 1Hz - 1/6 MHz

OUTPUT: (Audio Out)

Sampleauflösung : 8-14 Bit (Amiga ohne Audiohardware)  
Samplingrate : 5 kHz- 72kHz (Amiga ohne Audiohardware)

AUDIO RENDERING: (non realtime)

Sampleauflösung : 8/16 Bit  
Samplingrate : 5 KHz- 200 KHz

Max 128 Samples \* Max 4MB per Sample = Max 0.5GB (!) per Song

## 1.10 über Soundqualität

Was will Symphonie ?

- Symphoniemusik will CPU intensiv sein
- Symphonie ist nicht für Hintergrundmusik gedacht sondern will HAUPTAPPLIKATION sein

Symphonie wurde geschrieben, um alle vorhandenen CPU Cyclen möglichst in gute Soundqualität zu investieren.

Symphonie erreicht aufgrund einiger spezieller Algorithmen im Vergleich zu Konkurrenzprodukten bessere Soundausgabequalität, dies obwohl Symphonie die Amiga Audio Chips noch NICHT optimal ausnutzt. (vgl Symphonie PRO )

Wie erreiche ich mit Symphonie optimale Qualität ?

Voraussetzung ist: gutes Rohmaterial (Samples):

- nur die besten Samples verwenden
  - Samplingrate 44.1 kHz oder höher
  - die Samples sollten sauber geschnitten sein
  - verwende 16Bit(od besser) Samples
- 
- Bildschirmmodi: Samplingrate Bildschirmmodus (Screenmode Requester)

-----	↔
-----	-----
4kHz- 28kHz	alle
28kHz- 54kHz	DblNTSC,DblPAL, MULTISCAN, Euro72
54kHz-100kHz ->40kHz	nur mit veränderten Bildschirmmodi , 31kHz ↔ HINWEIS: benutze "Monitor Spec Ed" von ↔ Raul A. Sobon

#### Benutzung in Symphonie

- falls die weitverbreiteten LQ (low quality) Samples verwendet werden, die z.B. in Protracker Modulen zu finden sind, sollten diese in Symphonie mittels der Downsample Funktion entsprechend vorbereitet werden.
- Benutze Euro72 oder Multiscan ScreenModus (VGA oder Multiscan Monitor) und schalte die Qualität von 28kHz auf mind 44kHz

#### Zukunftscompatibel sein:

- Das Modulformat ist hardwareunabhängig
- viele Stimmen bewirken intern eine erhöhte Sampleauflösung:

Bsp: ein 1 Channel Modul mit 8Bit Samples 56kHz  
wird gespeichert als: NUR 8 Bit, 56kHz Sound

Bsp: ein 64 Channel Modul mit 8Bit Samples 56kHz  
wird gespeichert als:

14Bit, 56kHz Sound !!! (~ CD Qualität)

Bsp: ein 16 Channel Modul mit 16Bit Samples 100kHz  
wird gespeichert als: 20 Bit, 100kHz Sound

Schluss: Module mit vielen virtuellen Kanälen  
können schnell hohe Qualität erreichen.

Die tatsächliche (hörbare) Ausgabequalität wird somit meist durch die jeweils vorhandene Audiohardware STARK eingeschränkt, und kann erst durch Aufrüstung im Hardwarebereich der internen Qualität etwas angenähert werden.

#### Rauschen

-----

Ursache: D/A Wandler der Hardware hat eine zu geringe Auflösung  
Beseitigung : 16 Bit Soundkarte oder besser

Ursache: Samples haben eine zu geringe Auflösung  
Lösung : 16Bit Samples verwenden

Quantisierungsrauschen  
-----

Ursache: D/A Wandler der Hardware hat eine zu geringe Auflösung  
Beseitigung : 16 Bit Soundkarte oder besser

Charakter: Samples tönen auf allen Frequenzen stark metallisch  
Ursache: System Mixrate ist zu gering (GUI:System/Freq)

Beseitigung : GUI:System/Freq erhöhen

Charakter: Ein Teil der Samples tönt metallisch, der Rest tönt gut  
Ursache: Die metallisch klingenden Samples 1x oder 2x "Downsamplen"  
(GUI:Instrument/D)

Allgemeine Beseitigung: MENU:Prefs/S. Preprocessor/Oversample Wert ↔  
erhöhen

Charakter: Dasselbe Sample tönt auf verschiedenen Tonlagen unterschiedlich  
stark metallisch

Ursache: Samplefrequenz des Samples ist zu gering

Lösung : Das Sample 1x oder 2x "Downsamplen" (GUI:Instrument/D)  
Um die ursprüngliche Tonhöhe beizubehalten ist die Instrumentstimmung  
um eine Oktave zu erhöhen

## 1.11 Einschränkungen der Pre Release

<REMOVED>

## 1.12 GUI Hilfe : Inhaltsverzeichnis

Information zu welcher Gruppe :

System Control    Systemfeld (inkl Dsp Kontrollfeld)

Instrument    Instrument Feld/ Samples laden

Waveform    Wellenform Graph/ Loopbereich markieren

Song    Song Struktur festlegen

Sequence    Sequenzfolge festlegen

Position    Patternfolge festlegen

---



PatternEd Feld Hauptfeld ( Editieren von Patterns)  
NoteEd Editierfeld für ein einzelnes Ereignisse

-----

Menu Information über Menubefehle

Tastaturbelegung

## 1.13 Menu Verzeichnis

Information zu welcher Menugruppe:

-----

? Auskunft über Symphonie

SYSTEM Player, Soundmodi, Dspmodi

FILE Allgemeine Dateioperationen (Song, Modul, Prefs)

EDIT Editieroperationen

MOVE Befehle für die Navigation

SAMPLE Sampleoperationen (inkl load/save)

PREFS Voreinstellungen

## 1.14 Info über : Menupunkt ?

ABOUT Information über das Programm  
(Version, Autor ...)

## 1.15 Info über : Menupunkt System

PLAYER -> x Spielt den entsprechenden Abschnitt/Teil des aktuellen  
Songs

PLAYER -> STOP Stopt den Songplayer

AUDIO MODE ->

OVERSAMPLE Ausgabetyp Oversampling Stereo 8 Bit einschalten  
8BT

OVERSAMPLE Ausgabetyp Oversampling Stereo 9 Bit einschalten  
9BT

Bemerkung: falls Oversampling eingeschaltet ist, wird die  
CPU ca 30-40% weniger(!) stark belastet.

STEREO 8BIT Ausgabetyp Stereo 8Bit einschalten

Bemerkung: 8Bit benötigt ca 1%-5% weniger Prozessorzeit  
als 9Bit Ausgabe

STEREO 9BIT Ausgabetyyp Stereo 9Bit einschalten

MONO Ausgabetyyp Mono 8Bit einschalten

MONO SURR Ausgabetyyp Mono 8Bit und Surround Modus einschalten

LEFT SURR Linke Audiokanäle in 9Bit und Surround Modus spielen

RIGHT SURR Rechte Audiokanäle in 9Bit und Surround Modus spielen

Bemerkung: SURROUND bewirkt, dass aus einem monophonen Signal künstlich  
ein stereophones Signal berechnet wird.

DSP -> DEACTIVATE alle DSP Effekte ausschalten

DSP -> ECHO DSP Echoeffekt ein-/ausschalten  
Dem Originalsignal wird ein Echo des Signals mit bestimmter  
Verzögerung und Stärke beigemischt.

DSP -> CROSSECHO DSP CrossEchoeffekt ein-/ausschalten  
Dem linken Kanal wird das Echo des rechten Kanals ↔  
beigemischt  
und umgekehrt.

DSP -> DELAY DSP Delayeffekt ein-/ausschalten  
Dem Originalsignal wird eine Verzögerung des Signals mit ↔  
bestimmter  
Verzögerung und Stärke beigemischt.

DSP -> CHORUS DSP Chorus Effekt ein-/ausschalten  
Dem Originalsignal wird eine leicht veränderte Kopie  
beigemischt.

QUIT Symphonie verlassen

## 1.16 Info über : Menupunkt File

### Songoperationen

-----

SONG -> LOAD Song laden  
SONG -> SAVE AS aktuellen Song unter Namen speichern  
SONG -> SAVE aktuellen Song wiederholt speichern

### Moduloperationen

-----

MODULE -> LOAD Song-Modul laden  
MODULE -> SAVE AS Song in Modulformat unter Namen speichern

Teile die gespeichert werden:

- \* Song
- \* alle Samples
- \* eine Textdatei (falls diese vorher geladen wurde)

MODULE -> DiZCRIPT ein Textfile laden und anzeigen. Die Textdatei wird beim spielen des Songs durch die request Funktion des Systems angezeigt. Die Datei muss möglichst kurz sein, und sie darf keine Sonderzeichen enthalten. ( <RETURN> ist erlaubt)

MODULE -> EXTRACT S alle Samples eines Modules ins t: Verzeichnis kopieren

Warnung: alle Samples im Speicher werden beim ausführen dieser Funktion gelöscht !!!

MODULE -> DELTA PACK alle Samples DELTAPACKed speichern

Bemerkung: (Deltapack wurde implementiert, um externen Kompressoren zu deutlich höheren Raten zu verhelfen Bsp: lha, PP, xpkSHRINK ...)

#### Prefs Operationen

-----

PREFS -> LOAD Laden einer Prefs Datei

PREFS -> SAVE AS Prefs Datei unter bestimmten Namen speichern

PREFS -> SAVE DEFAULT Aktuelles Setup als Startumgebung speichern

Flagwerte zur Steuerung der Prefs Dateioperationen

PREFS -> NO AUDIO Audio Umgebung (Beispiel Sample Boost, Stereo Setup) nicht verändern/speichern

PREFS -> NO VIDEO Video Umgebung (Fenster Anordnung) nicht verändern/ ↔ speichern

PREFS -> NO SONG SETUP Song Setup (Kanalzahl, Patternzahl, Patternlänge) nicht verändern/speichern.

Die obigen Flagwerte haben je nach Dateioperation verschiedene Bedeutung:

Ladevorgang (PREFS->LOAD) Das jeweilige Teilsetup soll beim Laden der ↔ Preferences Datei nicht verändert werden; auch dann nicht, wenn die ↔ Information in der Preferences Datei vorhanden wäre.

Speichervorgang (PREFS->SAVE) Das jeweilige Teilsetup soll nicht gespeichert ↔ werden.

## 1.17 Info über : Menüpunkt Edit

ALLGEMEINE BEFEHLE

X -> CUT           schneidet das betreffende Element heraus  
X -> COPY          kopiert das betreffende Element in den Buffer  
X -> PASTE         kopiert das Element des Buffers in die aktuelle  
Umgebung

#### SPEZIFISCHE BEFEHLE

PATTERN -> DUPLN/SHRINK/EXPAND   siehe Pattern

TRACK   -> MIRROR   aktuellen Track spiegeln  
TRACK   -> SWAP      aktuellen Track mit Buffer vertauschen  
TRACK   -> ROT UP    aktuellen Track nach oben verschieben  
TRACK   -> ROT DWN   aktuellen Track nach unten verschieben

BLOCK   -> ADD       Block hinzufügen  
BLOCK   -> EXPAND    Block erweitern  
BLOCK   -> CLEAR     Block löschen  
BLOCK   -> SWAP      Block im Buffer mit Block an der  
Crsr Position vertauschen.

NOTE     -> INSERT   Note einfügen  
NOTE     -> DELETE   Note löschen

-----  
MACRO -> RECORD      Aktiviert den Symphonie Ereignis Recorder

Ereignistypen, die aufgenommen werden:

- Tastatur
- Menuereignisse
- Gadgets

Information, die bei jedem Ereignis mit-  
aufgenommen wird:

- SHIFT Status

MACRO -> RePLAY     Spielt die aufgenommen Ereignisse ab

UNDO                letzte Veränderung "löschen"

## 1.18 Info über : Menupunkt Move

Die Move Befehle dem Navigieren und dem direkten Anwählen spezieller  
Orte in Symphonie.

SEQUENCE -> PREVIOUS vorherige Sequenz anwählen  
          -> NEXT    nächste Sequenz anwählen  
          -> FIRST   erste Sequenz anwählen  
          -> LAST    letzte Sequenz anwählen

POSITION -> PREVIOUS vorherige Position anwählen  
          -> NEXT    nächste Position anwählen

---

-> FIRST erste Position der aktuellen Sequenz anwählen  
 -> LAST letzte Position der aktuellen Sequenz anwählen  
  
 PATTERN -> PREVIOUS vorheriges Pattern anwählen  
 -> NEXT nächstes Pattern anwählen  
 -> FIRST erstes Pattern des Songs anwählen  
 -> LAST letztes Pattern des Songs anwählen  
  
 CRSR -> TOP LEFT Crsr zur ersten Note des Patterns bewegen  
 -> BOTTOM RIGHT Crsr zur letzte Note des Patterns bewegen  
 -> TOP Crsr zur ersten Zeile des Patterns bewegen  
 -> BOTTOM Crsr zur letzten Zeile des Patterns bewegen

## 1.19 Info über : Menüpunkt Sample

LOAD Sample laden  
  
 BANK -> LOAD Sampleauswahl laden  
 BANK -> SAVE AS Sampleauswahl speichern  
  
 KILL aktuelles Sample löschen  
  
 IMPORT -> 16BIT RAW 16Bit Sample laden (Rohformat)  
 EXPORT aktuelles Sample exportieren  
  
 VIRTUEL -> RECALC alle ViRT Samples neu berechnen  
 VIRTUEL -> MIX/QUE/CLONE  
  
 SUPPORT -> ADJUST für alle diskbasierten Samples einen neuen Filepfad setzen  
 PATH  
  
 SUPPORT -> RELOAD alle Samples neu laden.  
 ALL

## 1.20 Info über : Menüpunkt Prefs

Preferences:

SYSTEM -> SYSTEM BUFFER

Wertbereich (4..128)

Startwert 4

Wirkung Anzahl Buffer für das Multibuffering System von Symphonie  
 festlegen. Eine grosse Bufferzahl bewirkt eine gleichmässige  
 Nutzung der CPU. Bei einer kleiner Bufferzahl können sich  
 Spitzenbelastungen schnell(!) hörbar auswirken (Verzerrung)

Richtwerte: Samples antesten  
 Songs spielen

SYSTEM BUFFER : 4

SYSTEM BUFFER : 32 oder höher

SYSTEM -> DSP BUFFER

Wertbereich (1..255)  
Startwert 16  
Wirkung Maximale Bufferzahl für den Dsp festlegen

REALTIME -> NO POS CHANGE

Flag Wert  
Wirkung Beim Spielen des Songs soll die Positionsnummer/Patternnummer nicht aufgefrischt werden

REALTIME -> NO SCROLL

Flag Wert  
Wirkung Beim Spielen des Songs soll der Crsr des PatternEd nicht ↔ mitscrollen

REALTIME -> NO SPECTRUM

Flag Wert  
Wirkung Spektrum Analyzer deaktivieren

REALTIME -> NO SCOPE

Flag Wert  
Wirkung Sound Scope deaktivieren

REALTIME -> FORCE UPDATE

Flag Wert  
Wirkung Optische Instrumente (Beispiel: Scope) auch auffrischen, falls kein Song gespielt wird.

PATTED -> SET JUMPER

Flag Wert  
Wirkung Jumplänge für das Editieren im PatternEd festlegen

PATTED -> NO PITCH

Flag Wert  
Wirkung Beim Setzen von KeyOn Befehlen durch die Klaviatur soll die vorhandene Tonhöhe nicht überschrieben werden.

PATTED -> NO INSTRUMENT

Flag Wert  
Wirkung Beim Setzen von KeyOn Befehlen durch die Klaviatur soll die vorhandene Instrumentnummer nicht überschrieben werden.

PATTED -> NO VOLUME

Flag Wert  
Wirkung Beim Setzen von KeyOn Befehlen durch die Klaviatur soll das vorhandene Volumen nicht überschrieben werden.

LOAD SONG/MOD -> KEEP PATTERN NUMBER

Flag Wert  
Wirkung Die aktuelle Patternzahl soll beibehalten werden

---

Anwendung: Einen Song von 64 Patterns auf 100 Patterns erweitern ↔  
.

LOAD SONG/MOD -> CONVERT SONG

Flag Wert

Wirkung      Beim Laden eines Songs die aktuelle Songstruktur  
                 (Stimmzahl und Patternlänge) beibehalten

Anwendung: Einen Song von 8 auf 16 Stimmen erweitern.  
                 Einen Song "verlängern" (Patternlänge erhöhen)

S.PREPROCESSOR -> SET ANTIKNACK

Wertbereich (1..1024) in Samples

Startwert    32

Ziel           Pegelsprünge bei schlecht geschnittenen Samples automatisch  
                 und intelligent(!) entfernen.

Wirkung      Fade in/Fade Out Länge des Samplepreprocessor festlegen  
                 Ein kleiner Wert bewirkt, dass alle Samples einen harten  
                 Anschlag "bekommen"

Der Antiknack Algorithmus schaltet sich selbstständig aus, wenn  
das bestimmte Sample schon sauber geschnitten ist.  
(Silence auf erstem und letztem Sample)

Bemerkung    Dieser Parameter wirkt nur beim Laden/Berechnen von Samples

S.PREPROCESSOR -> SET SAMPLE BOOST

Wertbereich (1..200%)

Startwert    100%

Wirkung      Festlegen der Maximallautstärke für Samples

Bemerkung    Falls durch DSP Effekte Übersteuerung auftritt, sollte dieser  
                 Parameter vermindert werden.

Dieser Parameter wirkt nur beim Laden/Berechnen von Samples

Warnung      Dieser Parameter darf erst ab 16 Stimmen höher als 100% gesetzt ↔  
                 werden.  
                 (8 Stimmen: 1-100%, ab 16 Stimmen 1-200%)

S.PREPROCESSOR -> SET OVERSAMPLE

Wertbereich (1..8) in Samples

Startwert    1

Wirkung      Vermindert das Quantisierungsrauschen niederfrequenter  
                 Samples

Bemerkung    Dieser Parameter wirkt nur beim Laden/Berechnen von Samples

STEREO CONTROL -> SET SAMPLE DIFF

---





Frequenz in Hz |<| |>| festlegen der System Mixrate  
 Speed in Bpm |<| |>| festlegen der System Geschwindigkeit

Time | Sequenztime / Songtime | zeigt die Spiellänge der aktuellen Sequenz  
 und des gesamten Songs

Bemerkung: falls der Song nur aus einer Sequenz besteht, so gilt immer:  
 Songtime:= Sequenztime

#### Songplayer:

SONG - Spielt das aktuelle Lied  
 SEQ - Spielt die aktuelle Sequenz  
 POS - Spielt das aktuelle Lied ab der aktuellen Position  
 PATT - Spielt das aktuelle Pattern  
  
 STOP - Stoppt den Player

#### Echo Control

Len |<| |>| festlegen der Dsp Echo Effektlänge in Systemzyklen  
 Level |<| |>| festlegen der Dsp Echostärke (100%, 50%, 25%, 12.5%... )

#### Delay Control

Len |<| |>| festlegen der Delayverzögerung in Systemzyklen  
 Level (log) |<| |>| Signalstärke des verzögerten Signal wählen (100%, 50%, 25%, 12.5%... ) ↵

#### Chorus Control

Depth |<| |>| festlegen der Effekttiefe in Samples  
 Speed |<| |>| festlegen der Effektgeschwindigkeit in 10er Zyklen

## 1.23 Song - Help

DEF: Ein Song (Lied) besteht aus einer bestimmten Anzahl  
 aufeinanderfolgender Sequenzen

REC Schaltet um zwischen Live / Record Betrieb

LOAD Lied laden

SAVE AS Aktuelles Lied speichern

SAVE Aktuelles Projekt (Song oder Module) noch einmal unter demselben  
 Namen speichern ↵

MOD Laden eines Moduls

KILL           Aktuelles Projekt löschen, Audiosystem neustarten

LEN            Patternlänge neu festlegen (die aktuellen Daten gehen verloren)

PATTERN        Patternzahl neu festlegen (die aktuellen Daten gehen verloren)

(Warnung: Bei ungeschickter Kombination obiger Werte kann  
          schnell Speichermangel auftreten -> Bildschirm blinkt rot)

## 1.24 Sequence - Help

DEF: Eine Sequenz besteht aus einer Reihe aufeinanderfolgender Positionen

# [<] [>]   - Aktuelle Sequenz wählen (0..63)

|X|       - aktuelle Sequenz in den Buffer kopieren  
          und aktuelle Sequenz auf Startwerte setzen

|C|       - aktuelle Sequenz in den Buffer kopieren

|P|       - aktuelle Sequenz aus dem Buffer kopieren

|DUP|      - aktuelle Sequenz auf die nächsthöhere duplizieren

|DEL|      - aktuelle Sequenz herauschneiden

|PLAY|     - Diese Sequenz soll gespielt werden

|SKIP|     - Diese Sequenz soll übersprungen werden (Mute)

|END|      - Diese Sequenz markiert das Ende eines Liedes.  
          alle nachfolgenden Sequenzen werden nicht gespielt.

BGN    [<] [>]   - Wählt die Startposition der aktuellen Sequenz

LEN    [<] [>]   - Wählt die Anzahl Positionen, die gespielt werden sollen  
          (beginnend mit der Startposition der Sequenz)

TRNS   [<] [>]   - Legt einen Transponierwert für die aktuelle Sequenz fest

LOOP   [<] [>]   - Wählt wieoft diese Sequence wiederholt werden soll

## 1.25 Position - Help

DEF: Eine Position definiert, welcher Ausschnitt aus welchem Pattern  
      gespielt werden soll.

CPY       - kopiert die aktuelle Position in den Buffer

DUP       - kopiert die aktuelle Position auf die nächsthöhere  
          (alles folgenden Positionen werden um eine Position versetzt)

PST       - Schreibt die Position im Buffer in die aktuelle Position

CLR       - Setzt die aktuelle Position auf Standardwerte

INS       - alle folgenden Positionen werden um eine Position versetzt

- [<] [>] - Wählt die aktuelle Position
- DEL - löscht die aktuelle Position  
(alle folgenden Positionen werden um eine Position rückversetzt)

Parameter einer bestimmten Position:

- BGN - Startzeile des gespielten Patterns
- LEN - Zahl der zu spielenden Patternzeilen
- CYCL - Setzt die Abspielgeschwindigkeit relativ zur SYSTEM Geschwindigkeit für die aktuelle Position

Formel:  $\text{Reale BPM} = \text{SYSTEM\_BPM} \times 1 / \text{POSITION\_CYCL}$

Bsp: Cycl = 1 : maximale Abspielgeschwindigkeit (System Speed)  
Cycl = 3 : 1/3 der maximalen Abspielgeschwindigkeit  
Cycl = 5 : 1/5 der maximalen Abspielgeschwindigkeit

- TRNS - Definiert für die aktuelle Position einen Transponier-Wert  
( falls ein bestimmtes Instrument den NOTUNE Parameter aktiviert hat, wird dieses Instrument nicht transponiert )
- LOOP - Setzt fest, wie oft die aktuelle Positon geloopt (wiederholt) werden soll

## 1.26 Pattern - Help

DEF: Ein Pattern besteht aus einer bestimmten Anzahl Tracks abhängig von der im SYSTEM Menu gewählten Stimmenzahl.

## 1.27 Track

DEF: Ein Track ist die Grundeinheit des Patterns und beschreibt die Notenfolge für eine Stimme. Zwei aufeinanderfolgende Tracks (Bsp. Track 1 und 2) können als ein Stereo Track interpretiert werden.

## 1.28 Instrument

- [<] [>] - Wählt das aktuelle Instrument
- R - aktuelles Instrument noch einmal laden, bzw. aktuelles Instrument neu berechnen
- LOAD - Weist dem aktuellen Instrument ein bestimmtes Sample zu

VOL            - Lautstärke des aktuellen Instrumentes setzen.  
Wertbereich 1-200 (%)  
              100% = maximale Lautstärke, ohne dass Verzerrungen auftreten

#### Type (Instrument Typus setzen)

-----

1            - Standardinstrument ohne Loop ("1 Shot Sample")  
LOOP        - Endlos gelooptes Instrument

SUST        - Schaltet um auf Sustained (gehaltenes Instrument)  
Dieses Instrument spielt :

1. Den Anschlagsbereich des Samples
2. Anschliessend n-mal den Loopbereich des Samples  
(Die Loopzahl wird durch LOOP# bestimmt)
3. Schliesslich den Ausklingbereich des Samples

#### Op (Instrument Operatoren setzen)

\* (NOTUNE) - Ausschalten der Transponier Funktion der Position für  
das aktuelle Instrument

MUTE        - Stummschalten des aktuellen Instrumentes (Mute)

KILL        - Löscht das aktuelle Sample

              WARNUNG: Instrument die Quellsample für virtuelle Instrumente sind,  
sollten nicht gelöscht werden.

#### Virtuel (Virtuelle Instrumente kreieren)

-----

MIX        - Kreiert ein Virtuelles Instrument  
QUE        - Kreiert ein Virtuelles Instrument  
CLONE      - Kreiert ein Virtuelles Instrument

#### Tools (Samples manipulieren)

RVS    - Umkehroperation für das aktuelle Sample  
D    - Downsampleoperation für das aktuelle Sample  
FILT - Interpolationsfilter für das aktuelle Sample

#### Tune (Instrument stimmen)

FINE        - Feinstimmen des Instrumentes:  
              der Wertbereich -128 ... +127 umfasst ca. einen Ganzton

PITCH       - Halbton-stimmen des Instrumentes:  
              der Wertbereich -12 ... +12 umfasst 2 Oktaven

L (Loopzahl für "sustained" Instrument setzen)

über WAVEFORM

über Samples

über Virtuelle Instrumente

## 1.29 Waveform

- Visualisiert das Sample des aktuellen Instrumentes

- Dient der Wahl des Loopbereiches:

Durch Druck mit der LMB (linke Maus Taste) innerhalb der Waveform, ziehen mit der Maus (bei gedrückter LMB) und loslassen innerhalb der Waveform wird ein Loopbereich markiert.

Loopbereiche werden automatisch so optimiert, dass beim Loopübergang kein Pegelsprung ("Knacksen") auftritt.

## 1.30 NoteEd

### S I M P L E    E F F E K T E

Simple Effekte sind "stand-alone" Effekte, die keine Parameter besitzen und deshalb keine Variationen zulassen.

### K O M P L E X E    E F F E K T E

Komplexe Effekte haben einen(1) bis maximal drei(3) Parameter, so dass ihre Wirkungsweise je nach Parameter stark variieren kann.

---

#### PITCH EFFEKTE

---

SET	KOMPLEXER EFFEKT
PARAMETER	Pitch (C0..C5), Instr (0..127)
WIRKUNG	aktuelle Frequenz setzen

Bemerkung: die Instrumentnummer muss entsprechend gesetzt sein

D/U	KOMPLEXER EFFEKT
PARAMETER	Speed (0...255) bestimmt die Effektintensität
	0 bewirkt, dass der Pitchslide Effekt ausgeschaltet wird
WIRKUNG	die Tonhöhe soll stetig erhöht/vermindert werden

---

Bemerkung: KeyOn und SetPitch Ereignisse stoppen den Slideeffekt

ADD KOMPLEXER EFFEKT "PIT+"  
 PARAMETER Intensität (-128...127) bestimmt, wie stark  
 die Frequenz verändert wird  
 WIRKUNG Die aktuelle Frequenz des aktuellen Kanals um einen  $\leftrightarrow$   
 bestimmten  
 Wert ändern.

Bemerkung: Der Wertumfang von C (-128...127) entspricht ca vier Halbtönen

FORMEL: neue Frequenz := alte Frequenz \* (1 + C \* 1/1024)  
 := alte Frequenz + (alte Frequenz \* C / 1024)

VIBR KOMPLEXER EFFEKT  
 PARAMETER Speed (0..128) Geschwindigkeit des Vibratoeffekts  
 Rate (0..255) bestimmt wie stark sich der Effekt auf  
 die aktuelle Frequenz auswirken soll  
 (0= keine Wirkung, 255= maximale  $\leftrightarrow$   
 Wirkung)

WIRKUNG "Vibrato", die aktuelle Frequenz soll Sinusförmig schwingen

Bemerkung: Vibrato kann gleichzeitig mit anderen Effekten (Beispiel  $\leftrightarrow$   
 Pitchslide)  
 verwendet werden.

"+" SIMPEL EFFEKT PU1/PU2/PU4  
 WIRKUNG aktuelle Frequenz erhöhen

"-" SIMPEL EFFEKT PD/PD2/PD4  
 WIRKUNG aktuelle Frequenz vermindern

#### INSTRUMENT EFFEKTE

FROM KOMPLEXER EFFEKT  
 PARAMETER Sample Position (0..255) bestimmt die Startpositon im  $\leftrightarrow$   
 Sample  
 - Variante mit Pitch und Instr möglich  
 WIRKUNG Spielt das aktuelle Sample ab einer bestimmten  $\leftrightarrow$   
 Position.  
 Die aktuelle Frequenz wird beibehalten.

Bemerkung: Falls mit diesem Effekt zugleich ein bestimmtes Sample zu  $\leftrightarrow$   
 wählen  
 ist, muss die Tonhöhe gültig sein.

Die tatsächliche Sample Position errechnet sich aus der  $\leftrightarrow$   
 Summe

des Parameters und des Wertes von FADD

Beispiele:

Fall 1: FADD= 0

Parameter Sa.Pos(0...255) bestimmt direkt die Startpositon im Sample  
Bsp: 0=Anfang, 128=Mitte, 255=Schluss

Fall 2: FADD> 0

Parameter Sa.Pos(0...255) + FADD bestimmen die Startpositon im Sample

Bsp: FADD=5

C=0 : Anfang, C=128 : Mitte, C=250 : Schluss  
(0+5) (127+5) (250+5)

Bsp: FADD=128

C=0 : Mitte, C=127 Schluss, C=255 : Illegaler Wert da ↵  
ausserhalb  
(0+128) (127+128) (255+128 !) des ↵  
Samples

FR&P

KOMPLEXER EFFEKT

PARAMETER

Pitch (C0..C5) bestimmt die Tonhöhe  
Sample Position (0..255) bestimmt die Startpositon ↵  
im Sample  
(0=Anfang, 255=Ende)  
Instrument (0..127) bestimmt welches Instrument zu ↵  
spielen ist

WIRKUNG

"From & Set Pitch"  
wie FROM, zusätzlich wird noch die Tonhöhe und die  
Instrumentnummer entsprechend interpretiert

FSET

KOMPLEXER EFFEKT

PARAMETER

im Sample

Sample Position (0...255) bestimmt die Offsetpositon ↵

0 = Anfang, 128 = Mitte, 255 = Schluss

WIRKUNG

"From Add Parameter := "  
Setzt den FROMADD (Offset) auf den Wert von Sample ↵  
Position

Bemerkung: Die Wirkung dieses Befehl ist erst hörbar, wenn ein  
FROM oder FR&P Befehl verwendet wird.

FADD

KOMPLEXER EFFEKT

PARAMETER

Fine Pos C(-128...127) bestimmt, wie stark FADD  
verändert werden soll.

WIRKUNG

"From ADD Parameter verändern"  
Addiert zu FADD den Wert von Parameter C

Bemerkung: Die Kombination aus FAD= und FADD kann benutzt werden  
um sehr genau eine bestimmte Position im Sample  
anzuspringen

Die Wirkung dieses Befehl ist erst hörbar, wenn ein FROM oder FR&P Befehl verwendet wird.

FORMEL: Neuer FADD := alter FADD + C \* Samplelänge/16384

SVIB KOMPLEXER EFFEKT  
 PARAMETER Speed (1..127 ) in Systemzyklen, 0= statischer ↔  
 Effekt/Effekt halten  
 Rate (0..100%) Effekttiefe  
 WIRKUNG Lässt den Samplestart Pointer Sinusförmig rotieren  
 Dieser Effekt wirkt nur in Kombination mit den FROM, ↔  
 FR&P Befehlen

#### VOLUMEN EFFEKTE

-----

SET SIMPEL EFFEKT  
 PARAMETER Volumen (1..100%)  
 WIRKUNG Volumen des aktuellen Kanals setzen

D/U KOMPLEXER EFFEKT  
 PARAMETER Speed (0...255) bestimmt die Effekttintensität  
 0 bewirkt, dass der Volumenslide Effekt ↔  
 ausgeschaltet  
 wird  
 WIRKUNG das Volumen soll stetig vermindert/erhöht werden

VOL+ KOMPLEXER EFFEKT  
 PARAMETER Intensität(-128...127) bestimmt, wie stark  
 das Volumen verändert wird  
 WIRKUNG das aktuelle Volumen des aktuellen Kanals um einen ↔  
 bestimmten  
 Wert ändern.

FORMEL: neues Volumen := altes Volumen + C \* 1/8

Bemerkung: Falls das neue Volumen ausserhalb des erlaubten Bereichs zu liegen kommt, wird das alte Volumen beibehalten.

TREM KOMPLEXER EFFEKT "TREM"  
 PARAMETER Speed (0..128) Geschwindigkeit des Tremoloeffekts  
 Rate (0..255) bestimmt wie stark sich der Effekt auf  
 das aktuelle Volumen auswirken soll  
 (0= keine Wirkung, 255=maximale Wirkung ↔  
 )

WIRKUNG "Tremolo", das Volumen soll Sinusförmig schwingen

Bemerkung: Tremolo kann gleichzeitig mit anderen Effekten (Beispiel ↔  
 Volumenslide)  
 verwendet werden.



AC KOMPLEXER EFFEKT "EMPH", Shift of Emphasis

PARAMETER Start (0..100) Start Volumen (bei Samplestart)  
End (0..100) End Volumen (bei Sampleende)  
Type (0, 1) 0=Off, 1= RampTo

WIRKUNG Dieser Effekt, bewirkt, dass das Volumen direkt mit der  
Samplesposition im Sample gekoppelt wird. (SamplePosition To ↔  
Volume)

So können Anschlagsakzentuierungen oder Stereoeffekte  
erzielt werden.

Bemerkung: Alle Volumeneffekt ausser Tremolo, werden vorübergehend  
unhörbar gemacht, (wirken intern trotzdem weiter).

Um den Effekt auszuschalten muss der Typ 0 angewendet werden ↔  
.

OFF SIMPEL EFFEKT

PARAMETER KEINE

WIRKUNG Kanal stummschalten, Ausgabe des Samples sofort ↔  
stoppen

ON SIMPEL EFFEKT

PARAMETER KEINE

WIRKUNG Ausgabe des Samples weiterführen

#### WEITERE EFFEKTE

-----

Geschwindigkeit verändern

CYL= KOMPLEXER EFFEKT

PARAMETER Cyclen (1...255) bestimmt, wie schnell die folgenden  
Zeilen zu spielen sind

WIRKUNG Setzt die Abspielgeschwindigkeit neu.

Bemerkung: Die neue Geschwindigkeit wird beibehalten, solange nicht  
eine neue Position gespielt wird, oder die Geschwindigkeit  
durch Effekte verändert wird.

Bsp: CYCL = 1 : maximale Abspielgeschwindigkeit (System Speed)  
CYCL = 3 : 1/3 der maximalen Abspielgeschwindigkeit  
CYCL = 5 : 1/5 der maximalen Abspielgeschwindigkeit

KEYOFF SIMPEL EFFEKT

WIRKUNG                    Spielt den Ausklingbereich des aktuellen   ←  
                                  Instrumentes

Bemerkung: wirkt nur bei SUSTained Instrumenten

ECHO                    KOMPLEXER EFFEKT  
                                  PARAMETER                    EffektTyp (Off, Normal, CrossEcho)  
                                                                     Level (100%,50%,25%...) Echo Volumen  
                                                                     Länge (1..n) Effektlänge in Systemzyklen

WIRKUNG                    Die aktuelle Echo Umgebung setzen

DELY                    KOMPLEXER EFFEKT  
                                  PARAMETER                    EffektTyp (Off, Normal)  
                                                                     Level (100%,50%,25%...) Delay Volumen  
                                                                     Länge (1..n) Effektlänge in Systemzyklen

WIRKUNG                    Die aktuelle Delay Umgebung setzen

### 1.31 Pattern Editor Feld

In diesem Feld können Notenwerte eingegeben werden, dazu ist zuerst von LIVE auf RECORD Betrieb umzuschalten.

Die einzelnen Tracks sind folgendermassen arrangiert:

- 1. Track    Stereo Links
- 2. Track    Stereo Rechts
  
- 3. Track    Stereo Links
- 4. Track    Stereo Rechts
  
- 5. Track    Stereo Links
- 6. Track    Stereo Rechts

etc.

Ereignisse:

Ein Ereignis ist entweder ein Notenereignis,  
 ein Spezialeffekt oder ein komplexer Spezialeffekt.

Notenereignis : Notenformat

NNNNIIIVV    (N= Notenwert, I= Instrument Nr, V= Volume

Bsp:

C 2-00050    Instrument Nr 0 mit 50% Lautstärke(Volumen)  
                  und der Tonhöhe C Octave 2 spielen.

C#2-010+0 Instrument Nr 10 mit 100% Lautstärke(Volumen)  
und der Tonhöhe Cis Octave 2 spielen.

A#1-10001 Instrument Nr 100 mit 1% Lautstärke(Volumen)  
und der Tonhöhe Ais Octave 1 spielen.

Bemerkung: Ein Notenereignis bewirkt zudem, dass  
Pitchslide/Volumeslide Effekte auf Null gesetzt  
werden ( Inaktivierung)

Spezial(effekte) sind im Note Editor beschrieben.

Gadget Befehle:

[<] [>] - Wählt das aktuelle Pattern

BLOCKOPERATIONEN:

Blöcke setzen  
-----

Mark -> NOTE Note markieren  
Mark -> TRK den aktuellen Track markieren  
Mark -> PAT das aktuelle Pattern markieren  
Mark -> BLK Blockmarke an der aktuellen Crsr Position setzen  
(entspricht dem Druck auf die <RETURN> Taste)

Do / Grundlegende Blockoperationen  
-----

C Kopiert den markierten Block aus dem aktuellen Pattern  
in den Block Buffer

X Schneidet den markierten Block aus dem  
aktuellen Pattern.

P Kopiert den Block an die aktuellen Crsr Position

DUP Kopiert den aktuellen Block des aktuellen Patterns auf das  
nächsthöhere Pattern  
BEMERKUNG: die aktuelle Patternnummer wird um Eins erhöht

ADD fügt alle Noten aus dem Blockbuffer an der aktuellen Crsr Position ein  
( falls sich an einer bestimmten Stelle schon ein Ereignis befindet,  
so wird dieses beibehalten)

EXPAND fügt zwischen alle Zeilen des Blocks eine Leerzeile ein.  
(Warnung: die untere Hälfte des Blocks wird somit  
gelöscht)

SHRINK entfernt alle ungeraden Zeilen des aktuellen Blocks

---

(Umkehroperation von EXPAND)

Rot (Block rotieren)

D Block um eine Zeile nach unten rotieren. Die unterste Zeile wird somit an die Position der ersten Zeile des Blocks verschoben.

U Block um eine Zeile nach oben rotieren. Die oberste Zeile wird somit an die Position der letzten Zeile des Blocks verschoben.

Pitch (Tonhöhe)

-/+ erniedrigt/erhöht die Tonhöhe (Pitch) jeder Note des aktuellen Blocks. ( SHIFT gedrückt -> -/+ Oktave)

Vol (Lautstärke)

-/+ erniedrigt/erhöht das Volumen (Lautstärke) jeder Note des aktuellen Blocks. ( SHIFT gedrückt -> -/+ 10 Einheiten)

Instr (Instrument Nummer)

-/+ erniedrigt/erhöht die Instrumentnummer jeder Note des aktuellen Blocks. ( SHIFT gedrückt -> -/+ 10 Einheiten)

## 1.32 Tastaturbelegung

Tastaturbelegung:

ESC - Stoppt den Player

F5 - Spielt das ganze Lied

F4 - Spielt die aktuelle Sequenz

F3 - Spielt die aktuelle Sequenz ab der aktuellen Position

F2 - Spielt das aktuelle Pattern

F1 - Den aktuellen Track "Mute'en" stummschalten

F9 - Record/Live Modus

F10 - Laden eines Samples

TAB - zur nächsten Tabulator Position

(TAB) - Tabulator setzen

RETURN - Blockanfang(1x) bzw Blockende(2x) setzen

SPACE - aktuelle Note löschen

(SPACE)- aktuellen Track löschen

---

Numeric Keypad:

[ ] - Position ab/auf  
{ } - Pattern ab/auf

Ins - Note einfügen  
Del - Note killen

- - Volumen -1  
+ - Volumen +1

## 1.33 Sample

Sample: Allgemeine Hinweise

Symphonie erkennt automatisch:

RAW 8Bit Samples  
RAW 16Bit MONO Samples

IFF 8Bit MONO Samples (unpacked)  
IFF 8Bit STEREO Samples (unpacked)

WAVE 8Bit MONO (PC, unpacked)  
WAVE 16Bit MONO/STEREO (PC, unpacked)

MAESTRO 16BIT MONO/STEREO (Samplitude)

Weitere Fremdformate auf Anfrage.

Sample: Ladevorgang

Beim Laden eines Samples werden automatisch folgende Prozesse durchlaufen: (Sample Preprocessor)

1. Konvertieren nach 16Bit (falls kein 16Bit Sample)
2. Maximieren des Samples (Ausnutzung der Bit Bandweite)
3. Pre-Oversampling
4. Anti Knack (entfernt Amplitudensprüngen  
am Anfang und am Ende des Samples, )
5. Konvertieren nach 8Bit (Symphonie JR)

## 1.34 GUI: Gruppe - Begriffserklärung

Eine Gruppe ist eine optische/funktionale Ansammlung verschiedener Hilfsmittel und Werkzeuge. Eine Gruppe ist durch einen 3D Rahmen begrenzt und besitzt einen bestimmten Titel. zB. SYSTEM.

## 1.35 Virtuelle Instrumente - Begriffserklärung

Ein virtuelles Instrument wird durch die in Symphonie eingebauten Synthese-Funktionen kreiert. Syntheseverfahren ist : Wavetable-Synthesis.

Vorteile der virtuellen Samples:

- benötigen keinen Festplattenspeicher, da sie jedesmal wenn der Song/ das Modul geladen wird, automatisch neu berechnet werden.
- können benutzt werden um:
  1. Akkord-Samples zu kreieren
  2. Mix-Samples zu kreieren (Bsp: Hihat und BassDrum -> 1 Sample )
  3. mehrere Tracks auf einen Track zu reduzieren
  4. Samples filtern, Samples rückwärts abspielen, Chorus
  5. komplexe Spezialeffekte
  6. Qualität eines Samples steigern (-> Downsampling)

Um ein virtuelles Sample zu kreieren, ist im PattEd die entsprechende Tonfolge zu schreiben

Interpretiert wird von Symphonie:

1. Notenergebnis (Tonhöhe, Instrumentnummer, Volumen)  
Bsp: C-4 001+0      C-4      001      +0 (=100)
2. Notenergebnis mit PitchUp/PitchDwn Effekt

Bis jetzt sind drei virtuelle Typen implementiert:

MIX    Samples  
QUE    Samples  
CLONE   Samples

Parameter eines Samples:

RVS   Rückwärts spielen...  
D   Downsampling...  
FILT   Interpolationsfilter...

---

### 1.36 MIX (kreieren eines virtuellen Instrumentes)

Die Mix Funktion dient dem Zusammenmischen mehrerer Samples zu einem einzigen Sample.

Ausgangsposition ist die aktuelle Crsr Position. Dabei werden alle Samples einer Zeile ab Crsr zu einem neuen Sample gemixt.

1. Länge und Stimmlage des resultierenden Samples ist mit dem Sample der Ausgangsposition identisch

ACHTUNG: um kompatibel zu bleiben, ist das Volumen der zu mixenden Samples auf 100 zu setzen.

Bemerkung: Bei nur einem Sample wird ein CLONE erzeugt

### 1.37 MIX (kreieren eines virtuellen Instrumentes)

Die Que Funktion dient dem Aneinanderreihen mehrerer Samples zu einem einzigen Sample.

Ausgangsposition ist die aktuelle Crsr Position. Dabei werden alle Samples einer Spalte ab Crsr zu einem neuen Sample zusammengefügt.

1. Stimmlage des resultierenden Samples ist mit dem Sample der Ausgangsposition  $\leftrightarrow$  identisch
2. Länge des resultierenden Samples setzt sich aus der Summe der Teilsamples  $\leftrightarrow$  zusammen.

ACHTUNG: um kompatibel zu bleiben ist das Volumen der zu mixenden Samples auf 100 zu setzen.

Bemerkung: Bei nur einem Sample wird ein CLONE erzeugt

### 1.38 CLONE (kreieren eines virtuellen Instrumentes)

Die Clone Funktion macht eine virtuelle Kopie des aktuellen Samples. Das nächsthöhere Sample wird dabei überschrieben.

### 1.39 RVS (Parameter eines Samples)

Das aktuelle Sample wird gespiegelt

### 1.40 Downsample (Parameter eines Samples)

Diese Funktion verschiebt das aktuelle Sample um eine Oktave tiefer (mit SHIFT höher).

Die Zwischensamples werden linear interpoliert, somit wird die Samplingfrequenz verdoppelt

-> um die ursprüngliche Tonhöhe zu erhalten  
ist das Sample eine Oktave höher anzupspielen

-> das Quantisierungsrauschen niederfrequenter Samples  
vermindert sich stark (Qualitätssteigerung)

-> Pro Instrument kann diese Funktion MAX 127x angewandt  
werden. ( 1Byte -> 100000000000000000000000000000000 GByte !)

## 1.41 FILT (Parameter eines Samples)

Filtertypus: Interpolationsfilter

Mit diesem einfachen Filter, können Instrumente  
aufgehellert oder abgedunkelt werden (Brightness).

Ohne SHIFT: Ein Klick auf das FILT Gadget bewirkt eine Zunahme der Filtertiefe  
(abdunkeln)

Mit SHIFT: Ein Klick auf das FILT Gadget bewirkt eine Abnahme der Filtertiefe  
(aufhellen)

WertBeispiele (Filtertiefe) :

-2	-1	0	1	2
sehr hell	hell	FILTER INAKTIV	dunkel	sehr dunkel

HINWEISE: - Um die Filterfunktion auf normale (nicht virt.) Instrumente  
anzuwenden, ist von diesem zuerst ein CLONE zu erstellen.

- Mehrmaliges Abdunkeln/Aufhellen bewirkt keine(!) Qualitätsverluste.

---