

Symphonie - Help

COLLABORATORS

	<i>TITLE :</i> Symphonie - Help		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 11, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Symphonie - Help	1
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	3
1.5	Schema eines 96' Symphonie	4
1.6	Warum, Wo erhältlich ...	5
1.7	über Int. Money Order	6
1.8	über Cash	6
1.9	über Symphonie Innereien	6
1.10	über Soundqualität	7
1.11	Einschränkungen der Pre Release	9
1.12	Help : Inhaltsverzeichnis	10
1.13	Menu Verzeichnis	10
1.14	Info über : Menüpunkt ?	11
1.15	Info über : Menüpunkt System	11
1.16	Info über : Menüpunkt File	12
1.17	Info über : Menüpunkt Edit	12
1.18	Info über : Menüpunkt Move	13
1.19	Info über : Menüpunkt Sample	13
1.20	Blockoperationen	14
1.21	System - Help	15
1.22	Song - Help	15
1.23	Sequence - Help	15
1.24	Position - Help	16
1.25	Pattern - Help	17
1.26	Track	17
1.27	Instrument	18
1.28	Waveform	19
1.29	NoteEd	19

1.30	Pattern Editor Feld	22
1.31	Tastaturbelegung	23
1.32	Sample	24
1.33	GUI: Gruppe - Begriffserklärung	25
1.34	Virtuelle Instrumente - Begriffserklärung	25
1.35	MIX (kreieren eines virtuellen Instrumentes)	26
1.36	MIX (kreieren eines virtuellen Instrumentes)	26
1.37	CLONE (kreieren eines virtuellen Instrumentes)	27
1.38	RVS (Parameter eines Samples)	27
1.39	Downsample (Parameter eines Samples)	27
1.40	FILT (Parameter eines Samples)	27

Chapter 1

Symphonie - Help

1.1 Symphonie Help Manual

Willkommen zu Symphonie

Einführung

Qualität
über Amiga Soundqualität

Internes

Helpmenu
Gadgets, Menus, Tastatur ...

Beta Version
Einschränkungen

History

Mods
woher bekomme ich demo modules

Zukunft
von Symphonie ...

Copyright

1.2 Where to get new modules for symphonie

Neue Sound Modules können bezogen werden bei:

Marco Ege
 Rigiblickweg 5
 CH 6048 Horw
 Switzerland

Bisher erschienen von M. Ege

Songname	Modus	Länge(*) (unpacked)
"Breathless"	8 Channel	300KB
"Capella Remix #1"	8 Channel	1.8MB
"Capella Remix #2"	8 Channel	1.6MB
"Deep Vale"	8 Channel	700KB
"Endless Ways"	16 Channel	570KB
"Liquid Time"	8 Channel	370KB
"Logical Dreams"	8 Channel	300KB
"Rainbow Run"	8 Channel	500KB
"Revenge of Chapell"	16 Channel	700KB
"Signing Off"	8 Channel	900KB
"Virtual Harmonies"	16 Channel	910KB
"Wild Canyon"	8 Channel	330KB

(*) Bem: Symphonie Modules erreichen Packraten zw 50% - 90% !! (xpkshink/lha)

1.3 program history

Symphonie V1.1c (Intern V1.061)

 16BIT PREPROCESSOR - 8 BIT MIXER - 9BIT AUDIOOUT

```
;1.01 Detected: VBR is not used -> crash on MC68040
; Fixed: Memory Trashing
; Waveform GFX trashing
```

```
; New: ViRT samples
; MovePrevInstrument
; All Routines are AREA_ID aware
```

```
; Changed: DownInstr,UpInstr
```

```
;1.02 Fixed: VBR no longer crashing on MC68040
```

```
; New Mouse Bar BOOPSI gadget BarGadget
; Instrument Finetuning
; NoteEd: New Outfit
```

```
;1.04 Fixed: LINESAMPLE MEM FAKE REALLOC
; LINESAMPLE FLAGSTUFF
; KILLSAMPLE routinez
; QUESample: falsche LEN, da bei LEN pitch nicht beruecksichtigt wird
; Div Zero on 60Hz Amiga
; IF POSNUMB < MODULEPOSNUMB(LOADING...) -> MEMOVERFLOW
```

```

;      SHIFT pressed -> cannolonger select LOOPRANGE with mouse!!!

; New   LINESAMPLE : IpolFilter
;      POSITION   : Tune (PosTuneUp,)
;      INSTRUMENT OUTFIT
;      ForceUpdateActInstrType,Pitch/Vol Down/Up Track(Shift)
;      Sequence
;      GUI changings
;      Security

; Changed   PattEdDrawStatus,PlaySongPattern

; Detect    MIX erhaelt teils falsche SRC position

;V1.052 BOOLGADGETS, ERRORHANDLER
;V1.053 EXTRACT SAMPLES
;V1.054 BLOCK COMMANDS,MACRO RECORDING,AnalyzeMsg(V2.2)
;V1.055 AUTOLOOP, 16BIT VIRTUELSAMPLE!
;V1.056 KNACKs reduced, PITCHSLIDE, VOLSLIDE, BLOCKInstr u/d
; modularity, new copysample, more SHIFT support
;V1.057 Screenmode Requester, Fixed 16Bit Linesample mix memerror
; Oversampling Version included, Fontadjust V0.1
; (gadtooltest fixed nolonger via addglist!!!)

;V1.058 Fixed Sync Prob (new period relativ algorithms)
; WAV autoconvert algorithm (to use Windows 3.11 samples !!!)

;V1.059 Fixed Oversample, Fixed Error in Normal_FREQLIST
; Fixed SONGDATA_LOOPCOUNT ERROR for POS>64
; Fixed Track_Mirror
; Fixed Knack at Sample END with x>1 samples playing

; NEW   Downsample, RVS, IPolfilter auch auf NICHT ViRT Samples !

;V1.060 Fixed Knack at Sample END with x>1 samples playing
;      ; Max Frequenz is now 65KHz (Quality: 225)
;V1.0601 DrawDblBorder had an missing RP move com -> gfx error, mem trashing ←
;      ???? fixed ?

;V1.061 FIXED: Sample Extract bug on stereo samples
; NEW: Sample load and process algorithms are 16Bit !
; FIXED: |R|eload sample now reloads stereo samples correctly
; NEW: |R|eload sample does a recalc on ViRT samples
; NEW: Sample tune range expanded to -/+24 halftones (4 octaves)
; NEW: Antiknack algorithm rewritten (now does a 32 sample vfade)
; FIXED: Stereo samples reimplemented
; NEW: Sequencer FX "FromAdd", "FromSet", "SetSpeed"
; NEW: multiwindow handler implemented (prepared for new
;      gui interpreter)
; DEBUGGING : 040COPYBACK support

```

1.4 (C)opy Right

Symphonie (C) by Patrick Meng - all rights reserved

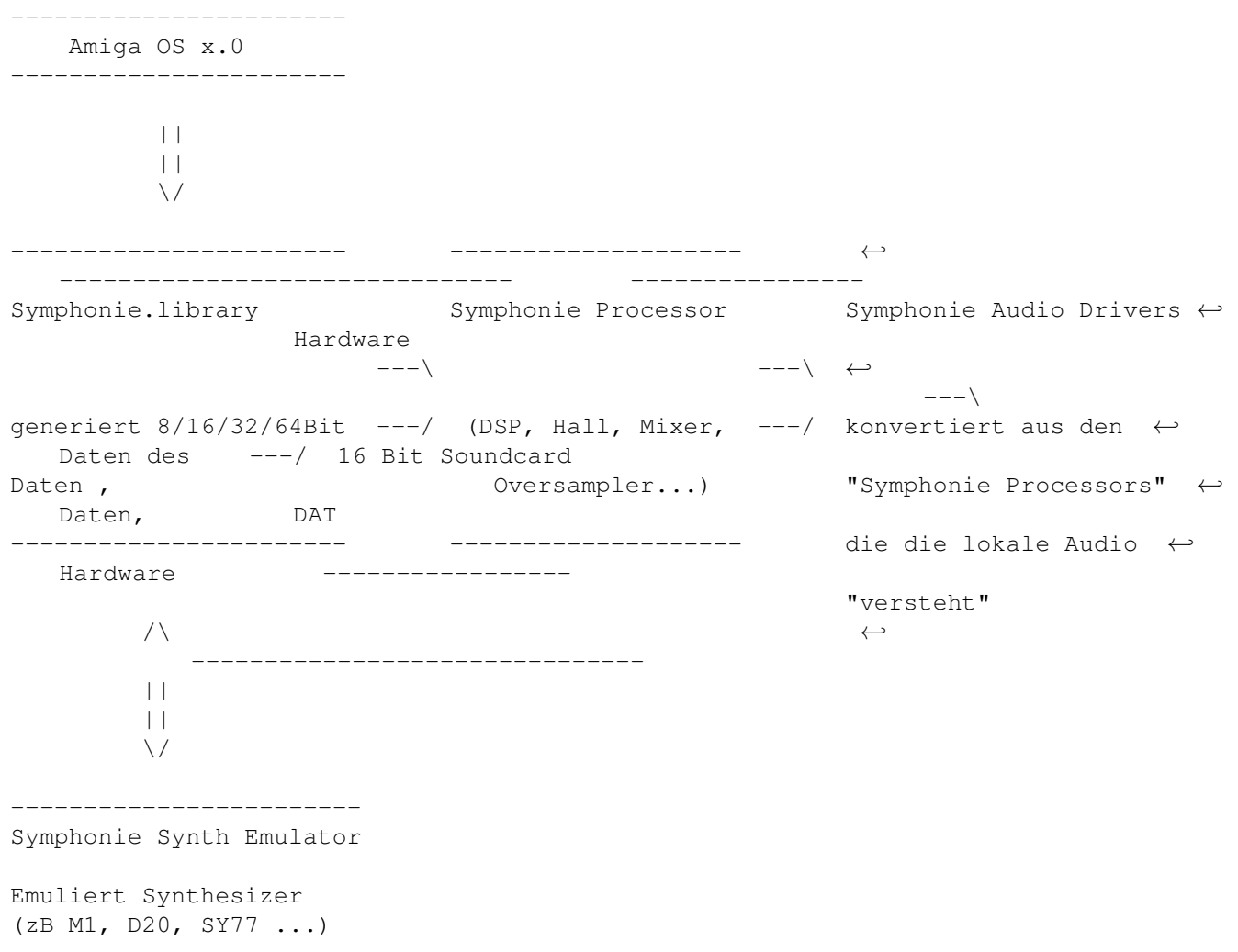
```
-----
Symphonie Demo (C) by Patrick Meng 1994
Symphonie Demo is FREeware. No further limitations.
-----
```

1.5 Schema eines 96' Symphonie

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

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

Aufbau:



1.6 Warum, Wo erhältlich ...

Symphonie ist ein Produkt, neuartiger 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 wird (Bsp 16Bit Soundkarten, DSPs, 16Bit über Parallelport, ev 16Bit Digital Out etc) Alle mit Symphonie geschriebenen Musikmodule werden von Symphonie automatisch an die jeweils vorhanden (Audio)Hardware optimal angepasste (Rauschen, Quantisierungsrauschen, Samplingrate, CPU Speed).

Symphonie gibt es in 3 Versionen

Symphonie 8BIT : 80 US\$ (Release Datum: out now !)

- 8...64 Soundchannels
- Intern 8Bit, Mixer 8Bit
- Audio OUT : 9Bit Stereo, 4kHz - 65kHz (!)

Symphonie PRO : 160 US\$ (Release Datum: Sommer/Fall 1994)

- 8...256 Soundchannels
- Intern 8/16Bit, Mixer 8/16/32Bit
- Audio OUT : 9Bit Stereo oder 12-14Bit ACM Stereo über Amiga
. 16Bit Stereo über Soundkarten Driver

Symphonie STUDIO :ca. 500+ US\$ (Release Datum Fall 1994)

- 8...256 Soundchannels
- Intern 8-32 Bit, mit FPU: 32Bit/64Bit
- 8-64Bit intern, Digital Hall Processor
- Audio OUT : 9Bit Stereo oder 12-14Bit ACM Stereo über Amiga
. 16-64Bit Stereo/Surround über (mehrere) Soundkarten

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

Available at:

Patrick Meng
Rosenfeldweg 4
6048 Horw
Switzerland

+41 41486948 (Info, Developer ...)

1.7 über Int. Money Order

Angaben:

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

(USA: via Citybank NY)

1.8 über Cash

Adresse:

Patrick Meng
Rosenfeldweg 4
CH-6048 Horw
Switzerland

1.9 über Symphonie Innereien

Symphonie wurde geschrieben in 68020 Assembler.

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

Die zentralen Routinen befinden sich schon in der 8. Generation und sind höchstoptimiert

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

Symphonie System Daten:

INPUT: (Samples)

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

OUTPUT: (Audio Out)

Sampleauflösung : 8-14 Bit (Amiga ohne Audiohardware)
Samplingrate : 5 kHz- 65kHz (Amiga ohne Audiohardware)

Volumenauflösung : 16Bit
 Frequenzauflösung : 48Bit (!)

Sample Preprocessor : 16Bit (IFILTER, DOWNSAMPLE, RVS)
 ViRT Sample Processor : 16Bit

2 Phase Oversampling
 Software DSP (falls keine Hardware DSP im System)

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: Requester)	Qualität	Samplingrate	Bildschirmmodus (Screenmode ↔
	-----	-----	↔
	20%-100%	5kHz-28kHz	egal (NTSC,PAL ... EURO36, SUPER72 ... MULTISCAN) ↔
	100%-185%	28kHz-54kHz	DblNTSC,DblPAL, MULTISCAN, Euro72
	185%-225%	54kHz-65kHz	nur mit veränderten Bildschirmmodi , 31kHz->40kHz ↔
			HINWEIS: benutze "Monitor Spec Ed" von Raul A. Sobon) ↔

Benutzung in Symphonie

- falls die weitverbreiteten LQ (low quality) Samples verwendet werden, die zB in Protracker Modulen zu finden sind, sollten diese in Symphonie mittels der Downsampling Funktion entsprechend vorbereitet werden.
- Benutze Euro72 oder Multiscan ScreenModus (VGA od Multiscan Monitor) und schalte die Qualität von 100(%) = 28kHz auf mind 180(%) = ca 52kHz
- benutze möglichst viele ViRT Samples (Akkorde, Wavetable Synthese)

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

- viele ViRT Samples bewirken intern eine erhöhte Sampleauflösung:

Bsp: ein 8 Channel Modul mit 8Bit Samples 56kHz

viele komplexe ViRT Samples mit ca
8fachem Overlay (8Samples -> 1ViRT Sample) werden
verwendet

Bem: Wer aus einem 8xOverlay ViRT Sample wiederum
8xOverlay Samples kreiert, erreicht somit
ein 64xOverlay. Dies entspricht einem 14Bit Sample (!)

Das resultierende Modul entspricht somit einem
8 Channel Modul mit 14Bit(od besser) Samples 56kHz

Es wird gespeichert als:

17Bit(od besser) , 56kHz Sound (besser als CD !)

Schluss: Module mit vielen Channels und vielen Virtuellen Samples 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 etwas angenähert werden.
(16/32Bit Soundkarten etc)

1.11 Einschränkungen der Pre Release

Einschränkungen der Pre-Release

System:

Keine Allokation der Audio Kanäle über audio.device
Komplette Übernahme der Audio Interrupts

"Sequenzen" NICHT voll implementiert

KEINE Symbiose der Pattern/Track/Note Funktionen mit
Block Funktion (Abstrahierung)

Fehler des GUI

Tastaturbelegung

KEIN 16 Bit IMPORT

KEINE Export Funktion

Interrupt Konflikte (Blitter, Seriel.port)

Dokumentation ist verbesserungswürdig (Informationsgehalt, Orthografie)

Funktionen:

wenige Funktionen NICHT implementiert/deaktiviert

Zukünftige Entwicklungsschwerpunkte (je nach Erfolg)

100% Windowsize adaptives und 100% Font sensitives GUI
Starke Veränderung des GUI (OS3.0 GUI Interpreter)

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

(Midi)

1.12 Help : Inhaltsverzeichnis

Welche
Gruppe
:

System

Song

Sequence

Pos

Pattern

Track

Block

Instrument

Waveform

Note Ed V1.2

Menu

Editor Feld

Tastaturbelegung

1.13 Menu Verzeichnis

Information zu welcher Menugruppe:

?

SYSTEM

FILE

EDIT

MOVE

SAMPLE

1.14 Info über : Menüpunkt ?

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

1.15 Info über : Menüpunkt System

PATTERN# neue Patternzahl festlegen (die aktuellen Daten gehen ↔
verloren)

PATTERNLEN neue Patternlänge festlegen (die aktuellen Daten gehen ↔
verloren)

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

DSP -> DEACTIVATE DSP wird ausgeschaltet

DSP -> SET DSP LEVEL DSP Effektintensität manipulieren

DSP -> ECHO DSP Echoeffekt einschalten
Effektintensität: 1= Echo ist laut
8= Echo ist sehr leise

DSP -> CROSSECHO DSP CrossEchoeffekt einschalten
Effektintensität: 1= Echo ist laut
8= Echo ist sehr leise

AUDIO MODE ->

OVERSAMPLE Ausgabetyyp Oversampling Stereo 8 Bit einschalten
BEM: falls OverSampling eingeschaltet ist wird die
CPU ca 30-40% weniger(!) stark belastet.

STEREO 8BIT Ausgabetyyp Stereo 8Bit einschalten
BEM: 8Bit benötigt ca 1%-5% weniger CPU Power 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 8Bit und Surround Modus spielen

RIGHT SURR Rechte Audiokanäle in 8Bit und Surround Modus spielen

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

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

PLAYER -> STOP Stopt den Songplayer

QUIT Symphonie verlassen

1.16 Info über : Menüpunkt File

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

MODULE -> LOAD Song-Modul laden
 MODULE -> SAVE AS Song in Modulformat 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

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

MODULE -> DELTA PACK alle Samples DELTAPACKed speichern

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

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 -> DUPLen/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
 Crsrposition vertauschen.

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

MODE -> PAINTINSTR falls dieser Modus aktiviert ist kann mit
 der <Space>taste das aktuelle Instrument
 gesetzt werden

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 : Menüpunkt Move

- nicht implementiert -

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	

Blocks. (SHIFT gedrückt -> -/+ 10 Einheiten)

1.21 System - Help

QUIT - Verlassen von Symphonie

Stimmzahl festlegen (Anzahl virtueller Audiokanäle)

8 - 8 Voices und System Neustart
16 - 16 Voices und System Neustart
32 - 32 Voices und System Neustart
64 - 64 Voices und System Neustart

Die 3 Folgenden Zahlen zeigen an:

1. System Qualität (10-185%) und System Samplingrate (5kHz..52kHz)
2. Player Speed absolut und in BPM (Beats pro Minute)

3. DSP Delay

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
TRACK - Spielt den aktuellen Track
STOP - Stoppt den Player

1.22 Song - Help

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

(IMPLEMENTIERT : 1 Sequenz)

SAVE AS - Aktuelles Lied speichern
SAVE - Aktuelles Lied noch einmal unter demselben Namen speichern
LOAD - Lied laden
SAVE MOD - Lied in Modulform speichern
(die Samples werden in Rohform ins Modul gepackt)
LOAD MOD - Laden eines Moduls

1.23 Sequence - Help

DEF: Eine Sequenz besteht aus einer Reihe aufeinanderfolgender Positionen

START [<] [>] - Wählt die Startposition des Liedes

-> aktiv wenn : das gesamte Lied wird gespielt

-> inaktiv wenn : es wird ab der aktuellen Position oder nur der/das aktuelle Track / Pattern gespielt

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

[Wertbereich : 1..512]

1.24 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 (alle 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 bst Position:

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

Formel: Reale BPM= SYSTEM_BPM x 1 / POSITION_SPEED

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

LOOP - Setzt fest wie oft die aktuelle Position geloopt (wiederholt) werden soll

TUNE - Definiert für die aktuelle Position einen Transponier-Wert

(falls ein bestimmtes Instrument den
 NOTUNE
 Parameter aktiviert
 hat, wird dieses Instrument nicht transponiert)

1.25 Pattern - Help

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

CPY - kopiert das aktuelle Pattern in den Buffer
 DUP - kopiert das aktuelle Pattern auf das nächsthöhere
 PST - Schreibt das Pattern im Buffer über das aktuelle Pattern
 CLR - löscht das aktuelle Pattern
 [<] [>] - Wählt das aktuelle Pattern

DUPLEN - Dupliziert den Ausschnitt, der beim Spielen des aktuellen Patterns
 ertönt, und hängt ihn am Schluss des Ausschnittes an.

Bsp: Die aktuelle Position zeigt:

```
START : 16   (Spielt 8 Zeilen: Zeile 16 bis Zeile 23 des
LEN   : 8    aktuellen Patterns )
```

Der Befehl DUPLEN kopiert somit Zeile 16 bis Zeile 23 des
 aktuellen Patterns auf die Zeilen 24 bis 31.

EXPAND - Fügt zwischen alle Zeilen eine Leerzeile ein.
 (Warnung: die obere Hälfte des aktuellen Patterns wird somit
 gelöscht)

SHRINK - Entfernt alle ungeraden Zeilen (Umkehroperation von EXPAND)

1.26 Track

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

CPY - kopiert den aktuellen Track in den Buffer
 X - extrahiert den aktuellen Track in den Buffer
 (führt die Befehle CPY und CLR aus)
 PASTE - Schreibt den Track im Buffer über den aktuellen Track
 CLR - löscht den aktuelle Track
 MIR - spiegelt den aktuellen Track
 ADD - fügt alle Noten aus dem Buffer im aktuellen Track ein
 (falls sich an einer bst Stelle schon eine Note befindet, wird

diese beibehalten, zudem werden keine Effekte aus dem Buffer übernommen.)

INS - Zeile an der Crsr Position einfügen
 DEL - Entfernt die Zeile auf der sich der Crsr befindet
 ROTD - Rotiert den ganzen Track um eine Position nach unten
 ROTU - Rotiert den ganzen Track um eine Position nach oben
 SWAP - Tauscht den aktuellen Track mit dem Buffer aus
 PITCH [<][>] - erniedrigt/erhöht die Tonhöhe jeder Note des aktuellen Tracks
 VOL [<][>] - erniedrigt/erhöht den Volumenwert jeder Note des aktuellen Tracks

CRSR Tasten - Wählt den aktuelle Track

1.27 Instrument

RECORD - Schaltet um zwischen Live / Record Betrieb

BANKS - Lädt ein komplettes Instrument Setup
 L - Speichert ein komplettes Instrument Setup
 KILL - Löscht das aktuelle Sample
 LOAD - Weist dem aktuellen Instrument ein bestimmtes Sample zu
 [<] [>] - Wählt das aktuelle Instrument

MIX
 - Kreiert ein
 Virtuelles Instrument

QUE
 - Kreiert ein
 Virtuelles Instrument

CLONE
 - Kreiert ein
 Virtuelles Instrument
 Instrument Parameter:

RVS
 - Umkehroperation für das aktuelle Samples

D
 - Downsampleoperation für das aktuelle Samples

FILT
 - Interpolationsfilter für das aktuelle Samples

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
- NOTUNE - Ausschalten der Transponier Funktion der
Position
für
das aktuelle Instrument
- X - Stummschalten des aktuellen Instrumentes (Mute)

Instrument stimmen:

- PITCH - Halbton-stimmen des Instrumentes:
der Wertbereich -12 ... +12 umfasst 2 Oktaven
- FINE - Feinstimmen des Instrumentes:
der Wertbereich -128 ... +127 umfasst ca. einen Ganzton

über

WAVEFORM
über
Samples
über
Virtuelle Instrumente

1.28 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.

1.29 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.

PITCH (C0...C5)

SET aktuelle Frequenz setzen
 (Bem das aktuelle Instr wird berücksichtigt !)
 P- aktuelle Frequenz vermindern
 P+ aktuelle Frequenz erhöhen

INSTRUMENT (0...127)

KEYOFF Spielt Ausklingbereich des aktuellen Instrumentes
 (Bem wirkt nur bei SUSTained Instrumenten)

VOLUMEN (0...100)

ohne Notenergebnis: aktuelles Volumen setzen
 OFF Ausgabe des Samples stoppen
 ON Ausgabe des Samples weiterführen

WEITERE EFFEKTE

DSPON DSP einschalten (falls DSP inaktiv -> keine Wirkung)
 DSPOFF DSP ausschalten

SPEEDUP erhöht den aktuellen (Pos-)Speed
 SPEEDDWN vermindert den aktuellen (Pos-)Speed

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.

Bei komplexen Effekten bekommen die Pitch/Instrument/Volumen Einstellungen neue Bedeutung:

Pitch = Parameter A
 Instrument = Parameter B
 Volumen = Parameter C

PITCH EFFEKTE

Pu/Pd Pitchslide Up/Down, Parameter C(0...255)
 Parameter C bestimmt die Effektintensität
 (falls C null ist, wird der Slideffekt gestoppt)

PIT+ Die aktuelle Frequenz des aktuellen Kanals um einen bestimmten Faktor ändern.

Parameter C(-128...127) bestimmt, wie stark die Frequenz verändert wird.

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

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

INSTRUMENT EFFEKTE

FROM Spielt das aktuelle Sample ab einer bestimmten Position.
Die aktuelle Frequenz wird beibehalten.

Parameter A, Parameter B(0...127), Parameter C(0...255)

Parameter A bestimmt, ob das aktuelle Instrument (Parameter B) zu berücksichtigen ist.

Bsp --- = Nicht berücksichtigen, sonst immer berücksichtigen

C-0 = berücksichtigen

D-1 = berücksichtigen

Parameter B bestimmt welches Instrument zu spielen ist

Parameter C(0...255) bestimmt die Startpositon im Sample

Fall 1: FADD= 0

Parameter C(0...255) bestimmt direkt die Startpositon im Sample

Bsp 0=Anfang, 128=Mitte, 255=Schluss

Fall 2: FADD> 0

Parameter C(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 "FRom & set P itch"

wie FROM, zusätzlich wird noch die Tonhöhe(Parameter A) entsprechend interpretiert

Parameter A(C0..C5), Parameter B(0...127), Parameter C(0...255)

Parameter A bestimmt die Tonhöhe

Parameter B bestimmt welches Instrument zu spielen ist

Parameter C(0...255) bestimmt die Startpositon im Sample

FAD= "From ADd Paramter := "

Setzt den FROMADD Befehl (Offset) auf den Wert von Parameter C

Parameter C(0...255) bestimmt die Offsetpositon im Sample

C=0 : Anfang, C=128 : Mitte, C=250 : Schluss

FADD "From ADD Parameter verändern"

Addiert zu FADD den Wert von Parameter C

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

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

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

VOLUMEN EFFEKTE

Vu/Vd Volumeslide Up/Down, Parameter C(0...255)
Parameter C bestimmt die Effektintensität

BEM: falls C null ist, wird der Volumeslideeffekt gestoppt

VOL+ Das aktuelle Volumen des aktuellen Kanals um einen bestimmten Wert ändern.

Parameter C(-128...127) bestimmt, wie stark das Volumen verändert wird.

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

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

WEITERE EFFEKTE

SPD= Setzt die Abspielgeschwindigkeit neu.

Parameter C(1...255) bestimmt, wie schnell die folgenden Zeilen zu spielen sind

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

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

1.30 Pattern Editor Feld

In diesem Feld könne 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.

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

Spezial(effekte) sind im
Note Editor
beschrieben.

1.31 Tastaturbelegung

General:

ESC - Stoppt den Player

- F5 - Spielt das aktuelle Lied
 - F4 - Spielt die aktuelle Sequenz
 - F3 - Spielt das aktuelle Lied ab der aktuellen Position
 - F2 - Spielt das aktuelle Pattern
 - F1 - Spielt den aktuelle Track
-

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

1.32 Sample

Sample: Allgemeine Hinweise

Symphonie erkennt automatisch:

- RAW 8Bit Samples
- IFF 8Bit MONO Samples
- IFF 8Bit STEREO Samples
- *.WAV Samples (8Bit MSDOS)

Die Release-Version von Symphonie wird noch

- RAW 16Bit MONO Samples
- RAW 16Bit STEREO Samples

unterstützen. Weitere Fremdformate sind denkbar.

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

1.33 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
.

1.34 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/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:

Notenereignis (Tonhöhe, Instrumentnummer, Volumen)

Bsp C-4 001+0 C-4 001 +0 (=100)

Notenereignis mit PitchUp/PitchDwn

Bis jetzt sind drei virtuelle Typen implementiert:

MIX
Samples

QUE
Samples

CLONE
Samples

Parameter eines Samples:

RVS
Rückwärts spielen...

D
Downsampling...

FILT
Interpolationsfilter...

1.35 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.

BEM: Bei nur einem Sample wird ein
CLONE
erzeugt

1.36 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 ↔ identisch
2. Länge des resultierenden Samples setzt sich aus der Summe der Teilsamples ↔ zusammen.

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

BEM: Bei nur einem Sample wird ein
CLONE
erzeugt

1.37 CLONE (kreieren eines virtuellen Instrumentes)

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

1.38 RVS (Parameter eines Samples)

Das aktuelle Sample wird gespiegelt

1.39 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 anzuspielen

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

-> Pro Instrument kann diese Funktion MAX 127x angewandt
werden. (1Byte -> 10000000000000000000000000000000 GByte !)

1.40 FILT (Parameter eines Samples)

Filtertypus: Interpolationsfilter

Mit diesem einfachen Filter, können Instrumente aufgehellt 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.
