

Das Kwave Handbuch

Thomas Eschenbacher
Deutsche Übersetzung: Sven-Steffen Arndt
:



Das Kwave Handbuch

Inhaltsverzeichnis

1 Einleitung	14
1.1 Kwave Ressourcen	15
1.2 Kwave - Historischer Rückblick	15
2 Grundlagen digitaler Audiotraktions	16
2.1 Die analoge Welt	16
2.2 Digitalisierung	17
2.3 Sample-Codierung	18
2.4 Sample-Formate	19
3 Kwave verwenden	21
3.1 Modus der Benutzerschnittstelle	21
3.2 Kommandozeile	23
3.2.1 Liste von Dateien oder Kommandos	23
3.2.2 GUI-Modus	23
3.2.3 Qt-Toolkit-Optionen	23
3.2.4 Einstellungen zum Programmstart	24
3.3 Öffnen und Speichern von Dateien	24
3.3.1 Unterstützte Dateiformate	24
3.3.2 Konvertierung zu und von .wav	25
3.3.3 Format von ASCII-Dateien	25
3.4 Erstellen einer neuen Datei	26
3.5 Aufnahme	27
3.6 Wiedergabe	28
3.7 Datei-Eigenschaften	30
3.8 Zoomen und Navigieren	30
3.8.1 Hinein- und Heraus-zoomen	30
3.8.2 Ansicht nach links und rechts verschieben	31
3.8.3 Verwenden der Übersicht	32
3.8.4 Vertikale Vergrößerung	32
3.9 Die Funktionsweise der Auswahl	32
3.9.1 Auswahl von Samples	32
3.9.2 Auswahl von Samples	32
3.10 Zwischenablage	33
3.11 Drag and Drop	34

Das Kwave Handbuch

4 Automatisieren und Skripten mit Kwave	35
4.1 Allgemeine Syntax	35
4.2 Verwenden der Befehlszeile	36
4.3 Kwave-Skripte	36
4.3.1 Allgemeine Struktur	36
4.3.2 Kommentare und Leerzeilen	37
4.3.3 Beenden	37
4.3.4 Markierungen	37
4.4 Befehlsreferenz	37
4.5 a	37
4.5.1 about_kde	37
4.5.1.1 Syntax: about_kde()	37
4.5.2 add_track	37
4.5.2.1 Syntax: add_track()	37
4.5.2.2 Siehe auch	38
4.6 c	38
4.6.1 clipboard_flush	38
4.6.1.1 Syntax: clipboard_flush()	38
4.6.2 close	38
4.6.2.1 Syntax: close()	38
4.6.2.2 Siehe auch	38
4.6.3 continue	38
4.6.3.1 Syntax: continue()	38
4.6.3.2 Siehe auch	38
4.6.4 copy	38
4.6.4.1 Syntax: copy()	38
4.6.4.2 Siehe auch	38
4.6.5 crop	39
4.6.5.1 Syntax: crop()	39
4.6.6 cut	39
4.6.6.1 Syntax: cut()	39
4.7 d	39
4.7.1 delayed	39
4.7.1.1 Syntax: delayed(Millisekunden, Befehl)	39
4.7.1.2 Parameter	39
4.7.1.3 Siehe auch	39
4.7.2 delete	39
4.7.2.1 Syntax: delete()	39
4.7.3 delete_track	40
4.7.3.1 Syntax: delete_track(Index)	40
4.7.3.2 Parameter	40
4.7.4 dump_metadata	40

Das Kwave Handbuch

4.7.4.1	Syntax: dump_metadata()	40
4.8	e	40
4.8.1	expandtolabel	40
4.8.1.1	Syntax: expandtolabel()	40
4.9	f	40
4.9.1	fileinfo	40
4.9.1.1	Syntax: fileinfo(Index)	40
4.9.1.2	Parameter	40
4.9.2	forward	41
4.9.2.1	Syntax: forward()	41
4.9.2.2	Siehe auch	41
4.10	g	41
4.10.1	goto	41
4.10.1.1	Syntax: goto(Position)	41
4.10.1.2	Parameter	41
4.11	i	41
4.11.1	insert_at	41
4.11.1.1	Syntax: insert_at(Position)	41
4.11.1.2	Parameter	41
4.11.1.3	Siehe auch	41
4.11.2	insert_track	42
4.11.2.1	Syntax: insert_track(Index)	42
4.11.2.2	Parameter	42
4.11.2.3	Siehe auch	42
4.12	l	42
4.12.1	label:add	42
4.12.1.1	Syntax: label:add(Pos[Text])	42
4.12.1.2	Parameter	42
4.12.2	label:delete	42
4.12.2.1	Syntax: label:delete(Index)	42
4.12.2.2	Parameter	43
4.12.3	label:edit	43
4.12.3.1	Syntax: label:edit(Index)	43
4.12.3.2	Parameter	43
4.12.4	loadbatch	43
4.12.4.1	Syntax: loadbatch(Dateiname)	43
4.12.4.2	Parameter	44
4.12.5	loop	44
4.12.5.1	Syntax: loop()	44
4.12.5.2	Siehe auch	44
4.13	m	44
4.13.1	menu	44
4.13.1.1	Syntax: menu (Befehl, Pfad, [Tastenkürzel], [ID])	44

Das Kwave Handbuch

4.13.1.2 Parameter	44
4.13.1.3 Unterbefehle	45
4.13.2 msgbox	47
4.13.2.1 Syntax: msgbox (<i>Text</i>)	47
4.13.2.2 Parameter	47
4.14 n	47
4.14.1 newsignal	47
4.14.1.1 Syntax: newsignal (<i>Samples, Abtastrate, Bits, Spuren</i>)	47
4.14.1.2 Parameter	47
4.14.2 next	48
4.14.2.1 Syntax: next()	48
4.14.2.2 Siehe auch	48
4.15 o	48
4.15.1 open	48
4.15.1.1 Syntax: open (<i>Dateiname</i>)	48
4.15.1.2 Parameter	48
4.15.1.3 Siehe auch	48
4.15.2 openrecent	48
4.15.2.1 Syntax: openrecent (<i>Dateiname</i>)	48
4.15.2.2 Parameter	49
4.15.2.3 Siehe auch	49
4.16 p	49
4.16.1 paste	49
4.16.1.1 Syntax: paste()	49
4.16.1.2 Siehe auch	49
4.16.2 pause	49
4.16.2.1 Syntax: continue()	49
4.16.2.2 Siehe auch	49
4.16.3 playback_start	49
4.16.3.1 Syntax: playback_start()	49
4.16.4 plugin	50
4.16.4.1 Syntax: plugin (<i>Name, [Parameter ...]</i>)	50
4.16.4.2 Parameter	50
4.16.4.3 Siehe auch	50
4.16.5 plugin:execute	50
4.16.5.1 Syntax: plugin:execute (<i>Name, [Parameter ...]</i>)	50
4.16.5.2 Parameter	50
4.16.6 plugin:setup	50
4.16.6.1 Syntax: plugin:setup (<i>Name, [Parameter ...]</i>)	50
4.16.6.2 Parameter	50
4.16.7 prev	51
4.16.7.1 Syntax: prev()	51

Das Kwave Handbuch

4.16.7.2 Siehe auch	51
4.17 q	51
4.17.1 quit	51
4.17.1.1 Syntax: quit()	51
4.17.1.2 Siehe auch	51
4.18 r	51
4.18.1 redo	51
4.18.1.1 Syntax: redo()	51
4.18.1.2 Siehe auch	51
4.18.2 redo_all	51
4.18.2.1 Syntax: redo_all()	51
4.18.2.2 Siehe auch	52
4.18.3 reenable_dna	52
4.18.3.1 Syntax: reenable_dna()	52
4.18.4 reset_toolbars	52
4.18.4.1 Syntax: reset_toolbars()	52
4.18.5 revert	52
4.18.5.1 Syntax: revert()	52
4.18.6 rewind	52
4.18.6.1 Syntax: rewind()	52
4.18.6.2 Siehe auch	52
4.19 s	52
4.19.1 save	52
4.19.1.1 Syntax: save()	52
4.19.1.2 Siehe auch	53
4.19.2 saveas	53
4.19.2.1 Syntax: saveas([Dateiname])	53
4.19.2.2 Parameter	53
4.19.3 saveselect	53
4.19.3.1 Syntax: saveselect()	53
4.19.3.2 Siehe auch	53
4.19.4 select_gui_type	53
4.19.4.1 Syntax: select_gui_type(<i>Modus</i>)	53
4.19.4.2 Parameter	53
4.19.5 select_track:all	53
4.19.5.1 Syntax: select_track:all()	53
4.19.5.2 Siehe auch	54
4.19.6 select_track:invert	54
4.19.6.1 Syntax: select_track:all()	54
4.19.6.2 Siehe auch	54
4.19.7 select_track:none	54
4.19.7.1 Syntax: select_track:none()	54

Das Kwave Handbuch

4.19.7.2 Siehe auch	54
4.19.8 select_track:off	54
4.19.8.1 Syntax: select_track:off(<i>Index</i>)	54
4.19.8.2 Parameter	54
4.19.9 select_track:on	54
4.19.9.1 Syntax: select_track:on(<i>Index</i>)	54
4.19.9.2 Parameter	54
4.19.10 select_track:toggle	55
4.19.10.1 Syntax: select_track:toggle(<i>Index</i>)	55
4.19.10.2 Parameter	55
4.19.11 selectall	55
4.19.11.1 Syntax: selectall()	55
4.19.12 selectnext	55
4.19.12.1 Syntax: selectnext()	55
4.19.12.2 Siehe auch	55
4.19.13 selectnextlabels	55
4.19.13.1 Syntax: selectnextlabels()	55
4.19.13.2 Siehe auch	56
4.19.14 selectnone	56
4.19.14.1 Syntax: selectnone()	56
4.19.15 selectprev	56
4.19.15.1 Syntax: selectprev()	56
4.19.15.2 Siehe auch	56
4.19.16 selectprevlabels	56
4.19.16.1 Syntax: selectprevlabels()	56
4.19.16.2 Siehe auch	56
4.19.17 selecttoleft	56
4.19.17.1 Syntax: selecttoleft()	56
4.19.17.2 Siehe auch	56
4.19.18 selecttoright	57
4.19.18.1 Syntax: selecttoright()	57
4.19.18.2 Siehe auch	57
4.19.19 selectvisible	57
4.19.19.1 Syntax: selectvisible()	57
4.19.20 start	57
4.19.20.1 Syntax: start()	57
4.19.20.2 Siehe auch	57
4.19.21 stop	57
4.19.21.1 Syntax: stop()	57
4.19.21.2 Siehe auch	57
4.19.22 sync	57
4.19.22.1 Syntax: sync()	57

Das Kwave Handbuch

4.19.22.2 Siehe auch	57
4.20 u	58
4.20.1 undo	58
4.20.1.1 Syntax: undo()	58
4.20.1.2 Siehe auch	58
4.20.2 undo_all	58
4.20.2.1 Syntax: undo_all()	58
4.20.2.2 Siehe auch	58
4.21 v	58
4.21.1 view:scroll_end	58
4.21.1.1 Syntax: view:scroll_end()	58
4.21.1.2 Siehe auch	58
4.21.2 view:scroll_left	58
4.21.2.1 Syntax: view:scroll_left()	58
4.21.2.2 Siehe auch	58
4.21.3 view:scroll_next	59
4.21.3.1 Syntax: view:scroll_next()	59
4.21.3.2 Siehe auch	59
4.21.4 view:scroll_next_label	59
4.21.4.1 Syntax: view:scroll_next_label()	59
4.21.4.2 Siehe auch	59
4.21.5 view:scroll_prev	59
4.21.5.1 Syntax: view:scroll_prev()	59
4.21.5.2 Siehe auch	59
4.21.6 view:scroll_prev_label	59
4.21.6.1 Syntax: view:scroll_prev_label()	59
4.21.6.2 Siehe auch	59
4.21.7 view:scroll_right	60
4.21.7.1 Syntax: view:scroll_right()	60
4.21.7.2 Siehe auch	60
4.21.8 view:scroll_start	60
4.21.8.1 Syntax: view:scroll_start()	60
4.21.8.2 Siehe auch	60
4.21.9 view:zoom_all	60
4.21.9.1 Syntax: view:zoom_all()	60
4.21.10 view:zoom_in	60
4.21.10.1 Syntax: view:zoom_in([Position])	60
4.21.10.2 Parameter	60
4.21.10.3 Siehe auch	60
4.21.11 view:zoom_normal	61
4.21.11.1 Syntax: view:zoom_normal()	61
4.21.12 view:zoom_out	61

Das Kwave Handbuch

4.21.12.1 Syntax: view:zoom_out([<i>position</i>])	61
4.21.12.2 Parameter	61
4.21.12.3 Siehe auch	61
4.21.13 view:zoom_selection	61
4.21.13.1 Syntax: view:zoom_selection()	61
4.22 w	61
4.22.1 window:activate	61
4.22.1.1 Syntax: window:activate([Überschrift])	61
4.22.1.2 Parameter	62
4.22.2 window:cascade	62
4.22.2.1 Syntax: window:cascade()	62
4.22.3 window:click	62
4.22.3.1 Syntax: window:click(<i>Klasse, x, y</i>)	62
4.22.3.2 Parameter	62
4.22.4 window:close	62
4.22.4.1 Syntax: window:close(<i>Klasse</i>)	62
4.22.4.2 Parameter	62
4.22.5 window:minimize	63
4.22.5.1 Syntax: window:minimize	63
4.22.6 window:mousemove	63
4.22.6.1 Syntax: window:resize(<i>Klasse, x, y</i>)	63
4.22.6.2 Parameter	63
4.22.7 window:next_sub	63
4.22.7.1 Syntax: window:next_sub()	63
4.22.8 window:prev_sub	63
4.22.8.1 Syntax: window:prev_sub()	63
4.22.9 window:resize	63
4.22.9.1 Syntax: window:resize(<i>Klasse, Breite, Höhe</i>)	63
4.22.9.2 Parameter	64
4.22.10 window:screenshot	64
4.22.10.1 Syntax: window:screenshot(<i>Klasse, Dateiname</i>)	64
4.22.10.2 Parameter	64
4.22.11 window:sendkey	64
4.22.11.1 Syntax: window:sendkey(<i>Klasse, Tastenkürzel</i>)	64
4.22.11.2 Parameter	64
4.22.11.3 Siehe auch	64
4.22.12 window:title	65
4.22.12.1 Syntax: window:title()	65
4.22.13 window:title_vertical	65
4.22.13.1 Syntax: window:title_vertical()	65

Das Kwave Handbuch

5 Module	66
5.1 Modul-Referenz	66
5.2 about (Über Kwave)	66
5.3 amplifyfree (Frei Verstärken)	67
5.4 band_pass (Bandpassfilter)	68
5.5 codec_ascii (ASCII Codec)	69
5.6 codec_audiofile (Audiofile Codec)	69
5.7 codec_flac (FLAC Codec)	71
5.8 codec_mp3 (MP3 Codec)	71
5.9 codec_ogg (Ogg Codec)	72
5.10 codec_wav (WAV Codec)	73
5.11 debug (Debug-Funktionen)	73
5.12 export_k3b (Export als K3b-Projekt)	74
5.13 fileinfo (Datei-Informationen)	76
5.14 goto (Gehe zu Position)	77
5.15 insert_at (Einfügen an)	78
5.16 lowpass (Tiefpassfilter)	79
5.17 newsignal (Neue Datei)	80
5.18 noise (Rauschgenerator)	81
5.19 normalize (Normalisierer)	82
5.20 notch_filter (Kerbfilter)	82
5.21 pitch_shift (Tonverschiebung)	83
5.22 playback (Wiedergabe)	84
5.23 record (Aufnahme)	85
5.24 reverse (Umkehren)	86
5.25 samplerate (Abtastrate ändern)	87
5.26 saveblocks (Blöcke speichern)	87
5.27 selectrange (Bereich auswählen)	89
5.28 sonogram (Sonagram)	90
5.29 stringenter (Befehl eingeben)	92
5.30 volume (Lautstärke)	93
5.31 zero (Null-Generator)	94
6 Fragen und Antworten	95
7 Danksagungen und Lizenz	97
7.1 Haupt-Autoren	97
7.2 Haupt-Beitragende	97
7.3 Weitere Beitragende, Copyright-Halter und andere	98
7.4 Dank an	99
A Datei-Informationen	100

Tabellenverzeichnis

4.1 Übersetzungstabelle für URL-Kodierung	36
A.1 Liste der Datei-Info-Schlüsselworte	104

Zusammenfassung

Kwave ist ein einfacher Sound-Editor basierend auf KDE Frameworks 5.

Kapitel 1

Einleitung

Dies ist "Kwave", ein einfacher Sound-Editor basierend auf KDE Frameworks 5. Er enthält folgende Funktionalität:

- eine Benutzerschnittstelle die zwischen SDI-, MDI- und Unterfenster-Modus umgeschaltet werden kann
- einfaches Ausschneiden, Kopieren und Einfügen
- mehrstufiges Rückgängig/Wiederherstellen
- Markierung von Signalen
- Aufnahme-Funktionalität, inklusive Vorab-Aufnahme
- Wiedergabe via Qt, PulseAudio, ALSA und OSS
- Aufnahme via PulseAudio, ALSA und OSS
- MP3 Import und Export
- Ogg/Vorbis und Opus Import / Export
- FLAC Import und Export
- einige Analyse-Funktionen wie Sonogramm
- verwendet intern 24 Bit Festkommaformat für Sample-Daten
- frei einstellbare Sample-Raten
- Unterstützung für das Bearbeiten von mehrspurigen Dateien
- Wiedergabe von mehrspurigen Audiodateien (Audioausgabe wird zu Mono oder Stereo heruntergemischt)
- erweiterbar durch eine einfach zu benutzende Modul-Schnittstelle
- Import/Export weiterer Audio-Dateiformate durch [libaudiofile](#)

Wenn Sie interessiert was getan wurde oder was noch getan werden muss, dann schauen Sie in die Dateien `CHANGES` und `TODO`, die den Quelldateien beiliegen. Hilfe und konstruktive Kritik sind immer willkommen.

1.1 Kwave Ressourcen

Falls Sie mit den Entwicklern in Kontakt treten wollen, weiterführende Hilfe zur Benutzung von Kwave brauchen, Patches, Fehlermeldungen oder anderes mitteilen wollen, dann werden die folgenden Adressen von Interesse sein:

- Projekt-Homepage

Für Informationen über aktuelle Releases oder andere Informationen über dieses Projekt, besuchen Sie bitte die [Kwave Homepage](#).

- GIT-Repository

Es gibt auch ein GIT-Repository auf den KDE-Servern und eine Spiegelung auf [SourceForge](#), von dem Sie die Quellcodes der jeweils aktuellen Entwicklerversionen beziehen können. Für Anweisungen wie Sie auf das GIT-Repository zugreifen können, lesen Sie bitte das entsprechende Kapitel zum Thema [Erstellen aus GIT](#) in der Entwickler-Dokumentation. Sie können aber auch einfach die Quellen über das Web-Interface auf [KDE](#) und auf [SourceForge](#) einsehen.

1.2 Kwave - Historischer Rückblick

Dieses Projekt wurde von Martin Wilz im Sommer 1998 begonnen und wurde entwickelt und verbessert durch ihn und einige andere. Im November 1999 begann Thomas Eschenbacher einige kleine Fehler hier und da zu beheben und vertiefte sich so mehr und mehr in den Quellcode des Programms. Bis heute hat er beinahe alle Komponenten des Programms erweitert, neu geschrieben oder bearbeitet und eine Menge Zeit investiert um es zu verbessern.

Seit Kwave v0.8.0 ist die Änderungshistorie nicht mehr im Benutzerhandbuch enthalten. Wenn Sie also an einer vollständigen Liste der Änderungen interessiert sind, finden Sie die vollständige Historie hier: <https://invent.kde.org/multimedia/kwave/-/blob/master/CHANGES>. Alternativ können Sie sich auch über das [GIT Web-Interface](#) die Quellcodes einsehen.

Kwave Version v0.9.0 ist die erste Version die auf KDE-Servern ([kdereview](#)) und auf SourceForge-Servern bereitgestellt wird, gefolgt von v0.9.1, der ersten Version für KDE Frameworks 5.

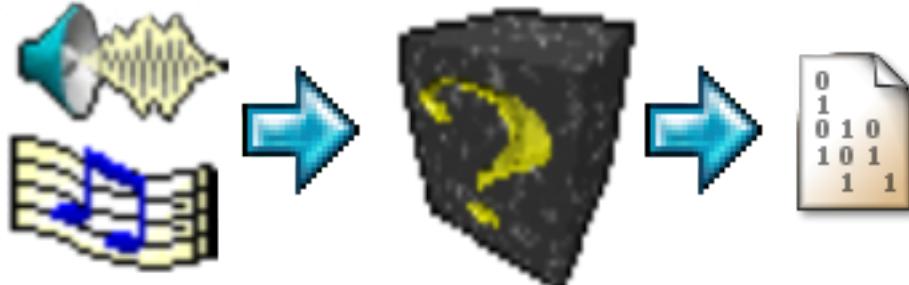
Kapitel 2

Grundlagen digitaler Audiotraktions

Dieses Kapitel soll eine kurze Einführung in die Grundlagen der digitalen Audiotraktions geben, ohne jedoch allzu sehr ins Detail zu gehen.

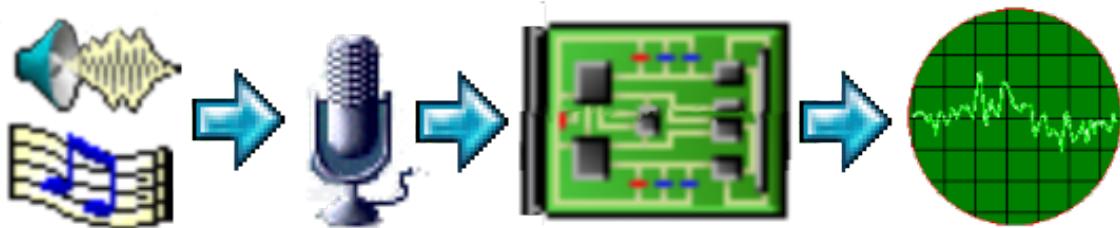
2.1 Die analoge Welt

Zuerst muss man sich im klaren sein, dass die Welt *analog* ist - Computer jedoch *digital* arbeiten. Es gibt also mehrere Wege um Audio von analog nach digital und umgekehrt umzuwandeln. Nachdem die Umsetzung von digital nach analog die Umkehrung der Umsetzung von analog nach digital ist, beschreiben wir nur letzteres.



Umwandlung von Sound in Bits

Bevor es weiter geht, muss analoges Audio zuerst in digitale Signale umgewandelt werden um in einem Computer verarbeitet werden zu können. Ein üblicher Weg dafür ist die Verwendung von einem Mikrofon und einem Verstärker. Diese Kombination hat Töne (also Luftdruckschwankungen) am Eingang und eine elektrische Spannung an ihrem Ausgang. Höhere Druckamplituden erzeugen dabei höhere Spannungen am Ausgang des Verstärkers. Der Verstärker produziert an seinem Ausgang ein sogenanntes 'Signal'. Anstelle eines Mikrofons kann man sich natürlich auch andere Audio-Quellen vorstellen. Und der „Verstärker“ kann derjenige sein, der in Ihrer Soundkarte eingebaut sein, wo man ihn normalerweise nicht sieht.



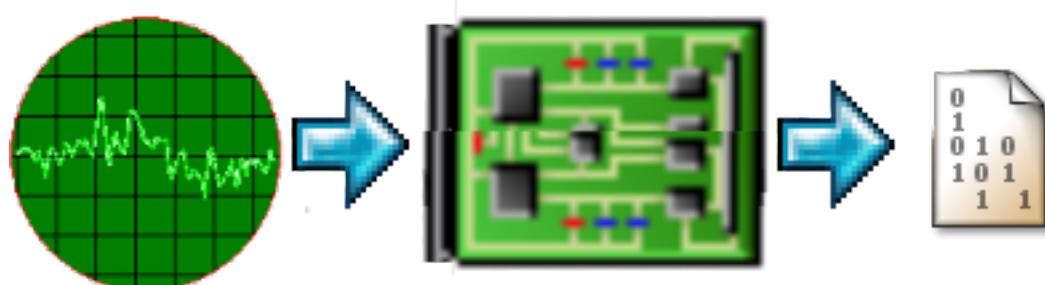
Umwandlung in ein elektronisches Signal

An dieser Stelle sollte man im Hinterkopf behalten, dass das elektrische Signal drei Beschränkungen unterliegt:

1. Die *Amplitude* ist auf einen Maximalwert beschränkt. Dies liegt daran, dass elektronische Verstärker nur Spannungen bis zu einer gewissen Grenze verarbeiten können. Das stellt kein Problem dar solange die Lautstärke nicht zu hoch wird. In diesem Fall würde das Signal sonst *begrenzt*, was bedeutet dass das Signal gegen seine Grenzen läuft und dadurch gestört wird.
2. Der *Frequenzbereich* ist ebenfalls begrenzt. Aufgrund der mechanischen Einschränkungen von Mikrofonen und dem begrenzten Frequenzbereich von Verstärkern ist auch der Frequenzbereich eines Signals begrenzt. Es gibt dabei keine harten Grenzen außerhalb derer das Signal abrupt verschwindet, aber unterhalb und oberhalb bestimmter Grenzen nimmt die Amplitude des Signals immer mehr ab. Die Existenz einer maximalen Frequenz lässt sich mit der begrenzten Geschwindigkeit erklären mit der ein elektrisches Signal steigen oder fallen kann. Indem man hochwertige Mikrofone und Verstärker verwendet, können diese Grenzen jedoch in Bereiche verschoben werden in denen sie vom menschlichen Ohr nicht mehr wahrgenommen werden können und daher nicht mehr interessieren. Das menschliche Ohr kann normalerweise keine Frequenzen oberhalb von 20 kHz mehr wahrnehmen.
3. Das Signal enthält *Rauschen*. Rauschen ist der unangenehmste Feind eines jeden der Audiosignale in irgendeiner Art und Weise bearbeiten muss. Rauschen ist ein typischer analoger Effekt, der das Audiosignal "unscharf" macht und stört, es ist immer vorhanden und kann nicht vermieden werden. Man kann lediglich versuchen, durch den Einsatz von qualitativ hochwertigen Komponenten so wenig Rauschen wie möglich zu erzeugen, so dass man es nicht hörbar wahrnehmen kann. Normalerweise besitzt das Rauschen eine gewisse Lautstärke, so dass das interessierende Signal im Vergleich zum Rauschen lauter sein sollte. Diese Relation wird auch als das *Signal-Rausch-Verhältnis (SNR)* bezeichnet. Je höher es ist, desto besser ist die Qualität des Signals. Töne die leiser sind als das Rauschen können nicht mehr gehört werden.

2.2 Digitalisierung

Wenn wir Audiosignale im Computer speichern oder abspielen wollen, müssen die analogen Klänge zuerst in digitale Daten umgewandelt werden. Dieser Vorgang wird als *Digitalisierung* bezeichnet. Es wandelt ein elektronisches Signal in eine Abfolge von digitalen Werten um.

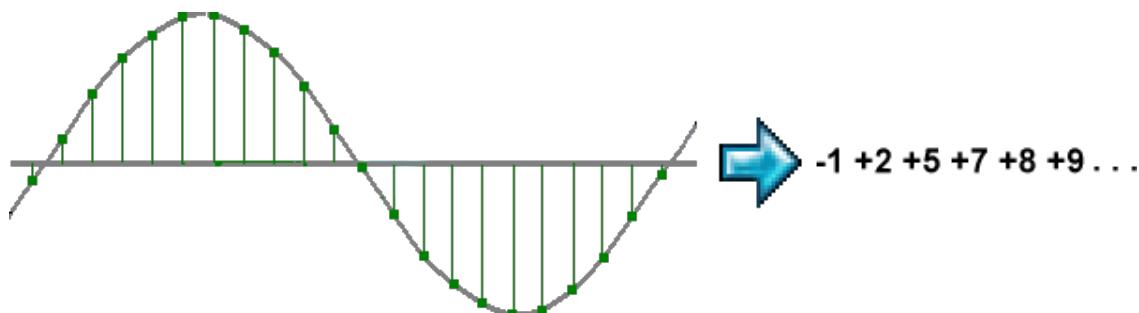


Digitalisierung des elektronischen Signals

Die Umwandlung kann als eine wiederholte Messung des augenblicklichen Wertes eines elektronischen Signals verstanden werden, es werden also einzelne Abtastwerte (Samples) bestimmt. Das Ergebnis wird dann als digitaler Wert kodiert.

Die Abtastung kann in wechselnden oder gleichmässigen Intervallen vorgenommen werden. Letztere Methode ist viel einfacher zu handhaben und wird deshalb normalerweise angewandt, mit einer konstanten Rate - der sogenannten *Abtastrate*. Übliche Abtastraten sind 8000, 11025, 22050 und 44100 Samples pro Sekunde. In der Praxis werden Abtastraten auch als Frequenzen in Hz oder kHz angegeben.

Die Abtastrate begrenzt die höchste Frequenz die digitalisiert werden kann. Aufgrund des Shannon'schen Abtasttheorems entspricht die höchste darstellbare Frequenz der halben Abtastfrequenz, somit können mit einer Abtastrate von 44,1 kHz keine Frequenzen oberhalb von ca. 22 kHz fehlerfrei digitalisiert werden. Um eine Verletzung dieser Halbe-Abtastrate-Regel zu vermeiden, besitzt Ihre Soundkarte bereits eingebaute Filter, die alle Frequenzen oberhalb der halben Abtastrate weg filtern.



Gesampeltes Signal

2.3 Sample-Codierung

Das Ergebnis der Abtastung ist eine Folge von *Samples* (Abtastwerten). Ein Sample repräsentiert den Wert eines Signals zu einem bestimmten Zeitpunkt als digitalen Wert.

Der Wert eines Samples kann auf verschiedene Arten interpretiert und kodiert werden. Die einfachste Art ist die *lineare* Kodierung. Dies bedeutet, dass der Wert eines Samples direkt dem Wert des Signals entspricht, multipliziert mit einem Faktor. Das ist zwar einfach zu handhaben, hat jedoch den Nachteil dass Rauschen besonders bei leisen Passagen zu hören ist, wo es am meisten stört, und weniger bei hohen Lautstärken, wo es weniger hörbar ist.

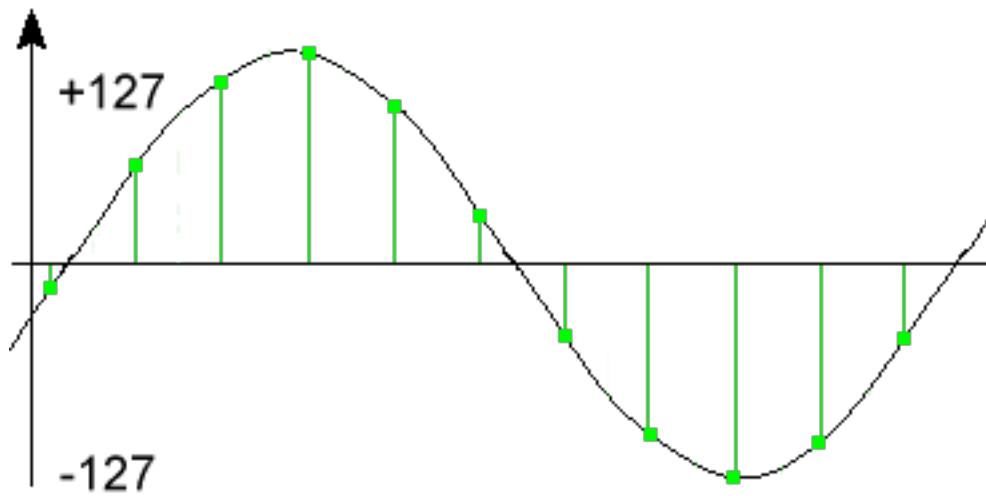
Eine Massnahme zur Reduzierung des Einflusses des Rauschens ist *nicht-lineare* Kodierung. Dies bedeutet, dass kleinere Amplituden vor der Verarbeitung angehoben werden. Indem kleinere Amplituden verstärkt werden, nimmt ihr Abstand zum Rauschen zu und die Qualität verbessert sich. Die beiden am weitesten verbreiteten Methoden sind *A-Law* und *U-Law* - zwei standardisierte logarithmische Verstärkungskurven, die in der digitalen Telefonie Verbreitung gefunden haben (ITU-Standard G.711).



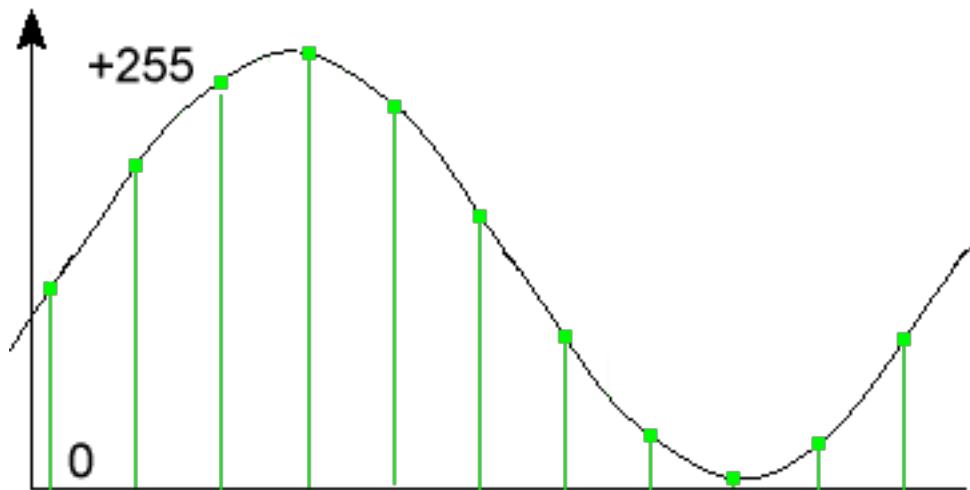
Nichtlineare Codierung

2.4 Sample-Formate

Samples können in verschiedenen Formaten und Genauigkeiten gespeichert werden. Die häufigsten sind Integer- (Festkomma-) Formate, die Werte in *festen* Quantisierungsstufen speichern. Abhängig davon wo die 'Null-Linie' festgelegt wird, muss man zwischen *vorzeichenlosen* (nur positive Werte, "Null-Linie" in der Mitte des Zahlenbereichs) und *vorzeichenbehafteten* (positive und negative Werte) Festkommaformaten unterscheiden.



Vorzeichenbehaftetes Format



Vorzeichenloses Format

Da die Quantisierung des Signals einem Verlust an Genauigkeit entspricht, verursacht sie sogenanntes *Quantisierungsrauschen*. Diese Art des Rauschens wirkt sich besonders auf leise Passagen aus, so dass diese Speicherformat nicht optimal ist, andererseits ist es aber äusserst einfach und schnell zu handhaben (Computer sind schnell bei Berechnungen mit Festkommazahlen).

Die zweite Möglichkeit besteht in der Verwendung von *Fliesskommazahlen*. Mit Fliesskommazahlen wird das Rauschen gleichmässig über alle Lautstärken verteilt und hat besonders bei kleinen Lautstärken grosse Vorteile. Leider ist dieses Format um einiges langsamer in der Verarbeitung (Computer sind bei Fliesskommazahl-Berechnungen im Vergleich zu Festkommazahl-Berechnungen um einiges langsamer).

Das Kwave Handbuch

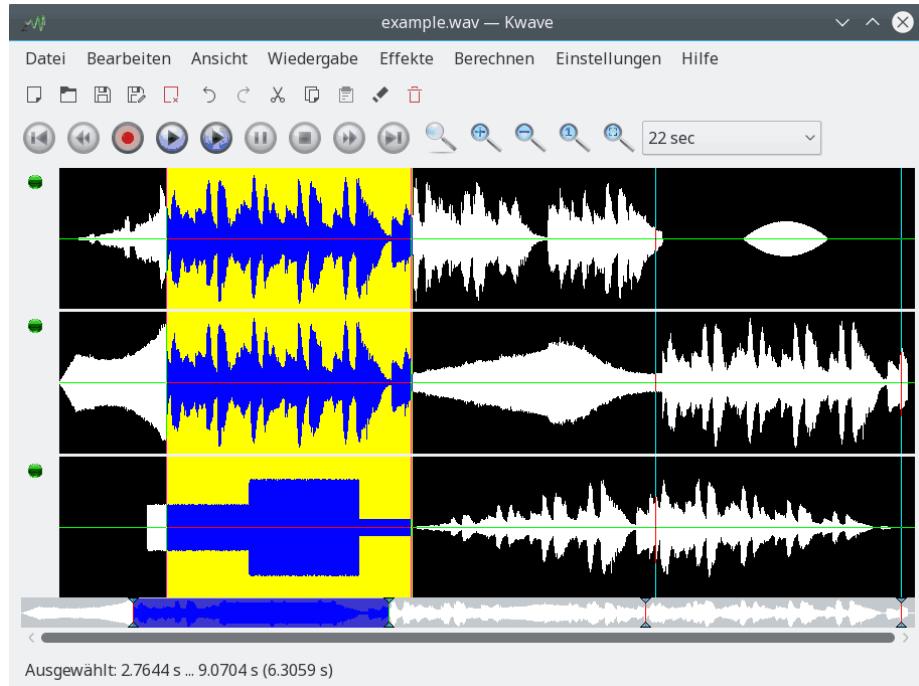
ANMERKUNG

Kwave verwendet intern ein Format mit *vorzeichenbehafteten Festkommazahlen* (Integer) mit 24 Bit Genauigkeit, gespeichert als 32-Bit Integer. Das hat zwar den Nachteil von hohem Speicherbedarf, besonders bei der Verarbeitung von Dateien mit geringerer Auflösung (z. B. 8 Bit), die Verarbeitung von 32-Bit Integer-Werten ist allerdings äußerst schnell und bietet noch einige Reserven, da nur 24 Bit davon genutzt werden.

Kapitel 3

Kwave verwenden

Hier ist ein Bildschirmfoto vom Kwave-Hauptfenster, um einen Eindruck zu bekommen, wie Kwave aussieht...



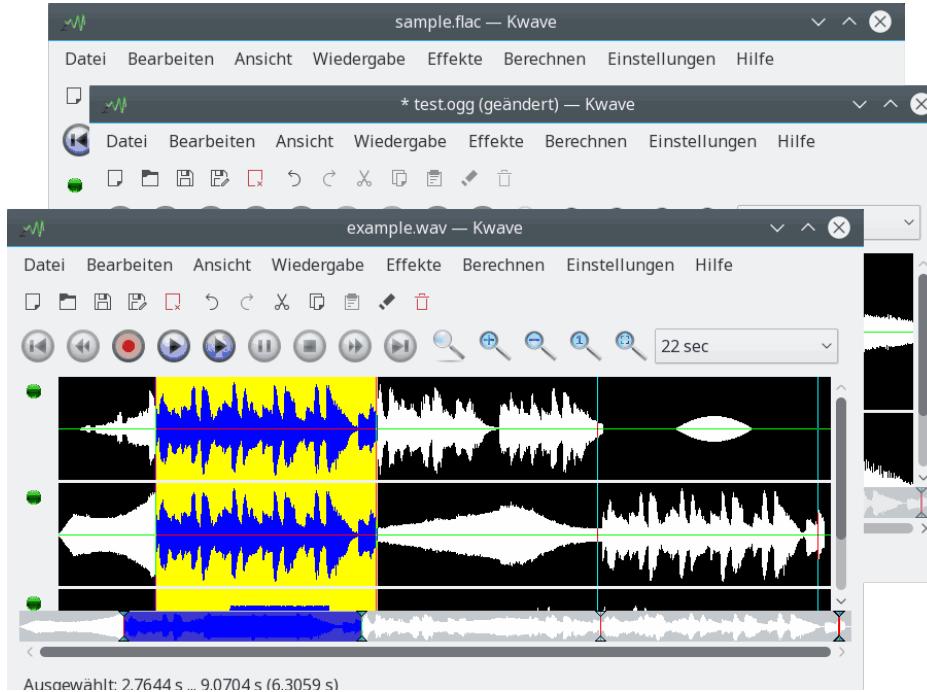
3.1 Modus der Benutzerschnittstelle

Sie können je nach Ihren persönlichen Vorlieben oder Anwendungsfällen einstellen wie sich Kwave beim Öffnen mehrerer Dateien verhält. Sie können diese Einstellung jederzeit über das Untermenü **Einstellungen → Zeige Dateien in...** vornehmen.

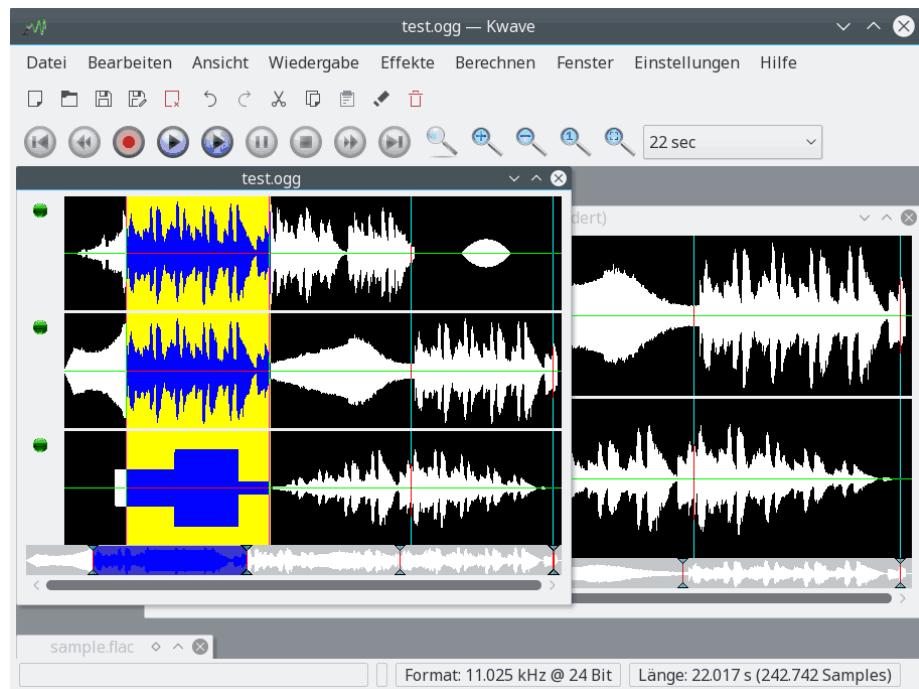
Die drei möglichen Einstellungen sind:

- **In separaten Fenstern (SDI):** Wenn Sie das *Single Document Interface* (SDI), verwenden, wird jede Datei in einem separaten Fenster angezeigt.

Das Kwave Handbuch

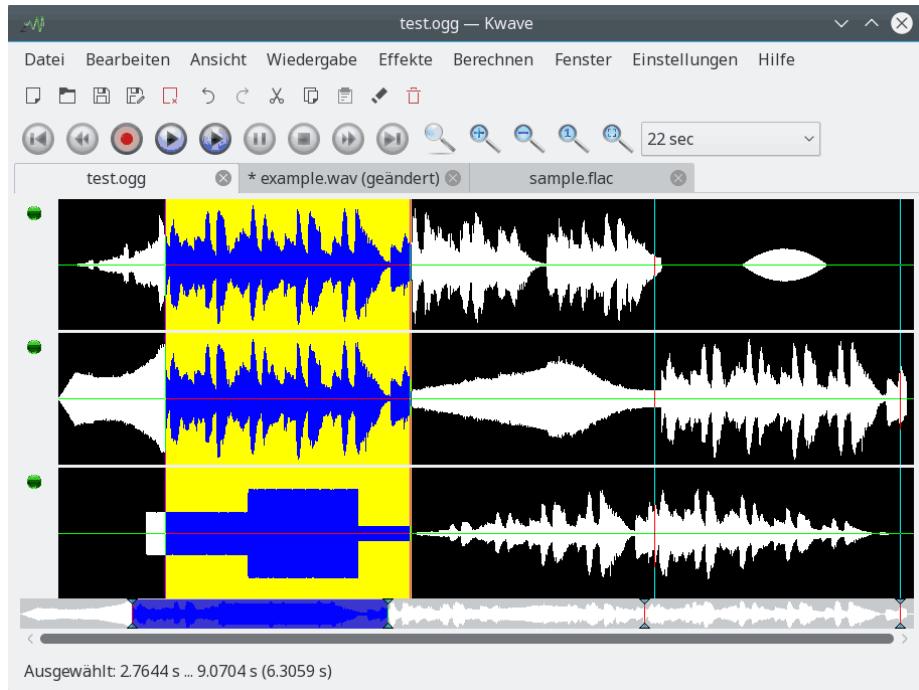


- **im gleichen Fenster (MDI):** Wenn Sie das *Multi Document Interface* (MDI), verwenden, dann gibt es nur ein einziges Kwave-Hauptfenster das jedoch einen Bereich enthält in dem Unterfenster angezeigt werden die Sie vergrößern, verkleinern, verschieben, staffeln oder minimieren können.
Sie können ein Fenster aus dem Menü **Fenster** auswählen oder mit **Strg-Tab** durch die Unterfenster wechseln.



- **Unterfenster:** Dies ist eine Variante des MDI-Modus, in dem Dateien in einzelnen Karteireitern dargestellt werden. Sie kennen diesen Modus wahrscheinlich von einigen gängigen Internet-Browsern.
Sie können ein Unterfenster über das **Fenster**-Menü aktivieren oder mit der Tastenkombination **Strg-Tab** zwischen den Unterfenstern wechseln.

Das Kwave Handbuch



3.2 Kommandozeile

3.2.1 Liste von Dateien oder Kommandos

Wenn Sie Kwave von der Kommandozeile aus starten, können Sie eine Liste von Dateien angeben, die geöffnet werden sollen. Die erste angegebene Datei wird dabei zuerst geöffnet und dann die anderen. Jede Datei wird dabei in einem neuen eigenen Fenster oder Unterfenster der selben Kwave-Instanz geöffnet. Wenn Sie Platzhalter-Zeichen verwenden, können Sie eine große Anzahl von Dateien auf einmal öffnen.

So startet zum Beispiel der folgende Befehl Kwave und öffnet alle Klangdateien des freedesktop Sound-Themes in jeweils einem eigenen Fenster oder Unterfenster:

```
% kwave /usr/share/sounds/freedesktop/stereo/*.ogg
```

folgenden Abschnitt-Befehle an Kwave übergeben, die als spezielle Adresse (URI¹) kodiert sind. Diese wird in einem [folgenden Abschnitt](#) beschrieben.

3.2.2 GUI-Modus

Der parameter `--gui=SDI | MDI | TAB` lässt Kwave in einem der drei möglichen GUI-Modi starten: [SDI](#), [MDI](#) or [TAB](#).

3.2.3 Qt-Toolkit-Optionen

Zusätzlich zu einer Dateiliste können Sie noch eine Liste von *Qt toolkit* Optionen wie `-qwindowgeometry` angeben, um die Grösse und/oder die Position des ersten geöffneten Kwave Fenster zu spezifizieren. Eine weitere Option ist `-display`, die Kwave auf einem anderen Bildschirm startet.

¹universal resource identifier

Der folgende Befehl startet zum Beispiel ein Kwave-Fenster mit einer Startbreite von 600 Pixeln und einer Höhe von 400 Pixeln, mit der rechten Seite 30 Pixel vom rechten Rand und 0 Pixel vom oberen Rand des Bildschirms entfernt.

```
% kwave --disable-splashscreen -qwindowgeometry 600x400-30+0
```

3.2.4 Einstellungen zum Programmstart

Mit dem Kommandozeilen-Parameter `--disable-splashscreen` können Sie das Startbild das beim Starten von Kwave erscheint deaktivieren. Dies kann nützlich sein wenn Sie Kwave von einem Shell-Skript aus starten wollen.

Der Kommandozeilen-Parameter `--iconic` lässt Kwave minimiert starten. Dies mag nützlich sein wenn Sie Kwave ohne sichtbare Benutzeroberfläche starten wollen, z. B. wenn Sie Kwave von einem Shell-Skript aus starten. Dieser Parameter deaktiviert implizit auch das Startbild!

Mit dem Befehlszeilen-Parameter `--logfile=kwave.log` können Sie die Abfolge der Aktionen einer Kwave-Sitzung in eine Datei speichern. Dies ist nützlich zur Fehlersuche, Sie werden möglicherweise nach einer solchen Log-Datei gefragt wenn Sie einen Fehlerbericht abgegeben haben.

3.3 Öffnen und Speichern von Dateien

Das Öffnen von Dateien mit Kwave funktioniert genauso wie in den meisten anderen Anwendungen auch, Sie können

- eine Liste von Dateien auf der [Kommandozeile](#) beim Start von Kwave eingeben
- ein leeres Kwave-Fenster öffnen (zum Beispiel mit **Datei → Neu... (Strg-W)**) und fügen eine Datei mittels [Drag and Drop](#) ein, oder Sie können
- eine Datei öffnen per Menü mit **Datei → Öffnen (Strg-O)**
- oder eine der zuletzt bearbeiteten Dateien auswählen unter **Datei → Zuletzt geöffnete Dateien**
- eine aktuelle Datei speichern mittels **Datei → Speichern (Strg-S)**,
- unter einem anderen Namen speichern mit **Datei → Speichern → Unter ... (Umschalt-Strg-S)**
- alle Bereiche die durch Marker begrenzt sind in jeweils eine eigene Datei speichern, mit **Datei → Speichern → Blöcke...**
- oder nur die aktuelle Auswahl mit **Datei → Speichern → Auswahl...**

3.3.1 Unterstützte Dateiformate

Kwave unterstützt folgende Dateiformate:

- Das favorisierte Dateiformat von Kwave ist (wie man vom Namen her bereits vermuten könnte) `*.wav`. Dieses Format ist weit verbreitet auf anderen "Betriebssystemen", sowie auch innerhalb der Plasma-Umgebung.
- Das zweite Format das Kwave unterstützt ist "ASCII". Sie können in ASCII sowohl exportieren als auch importieren. Bitte beachten Sie dass beim Speichern in diesem Format sehr große Dateien entstehen können! Das Dateiformat wird [im folgenden](#) beschrieben.
- `.mp3` und `.mp2` Import mit Hilfe der [libmad-Bibliothek](#) zum Importieren von MP3 in Kombination mit [id3lib](#) zur Decodierung von ID3-Tags und [lame](#) zum Exportieren.

- Ogg/Vorbis (*.ogg) Import und Export. Für Details besuchen Sie bitte <https://www.xiph.org>.
- FLAC (*.flac) Import und Export. Siehe <https://xiph.org/flac/> für weitere Details.
- Zusätzlich können Sie Dateiformate wie *.8svx (Amiga IFF/8SVX Sound File Format), *.au (NeXT, Sun Audio), *.aiff (Audio Interchange Format), *.avr (Audio Visual Research File Format), *.caf (Core Audio Format), *.nist (NIST SPHERE Audio File Format), *.sf (Berkeley, IRCAM, Carl Sound Format), *.smp (Sample Vision Format), *.snd (NeXT, Sun Audio), *.voc (Creative Voice) und andere mit Hilfe des [audiofile](#) Plugins importieren.

3.3.2 Konvertierung zu und von .wav

Im Moment ist es am besten, zum Bearbeiten von Formaten die nicht von Kwave unterstützt werden, ein externes Konverter-Programm zu benutzen. Eine gute Auswahl von Werkzeugen dafür finden Sie im [SoX](#) Paket, dort finden Sie auch gute Dokumentation!

Die Pläne für die Zukunft beinhalten Unterstützung für Importfilter für andere Formate und vielleicht einige Filter, die ein benutzerdefiniertes Skript mit einem Aufruf an externe Filter integriert, so dass sogar Formate gelesen und/oder geschrieben werden können, die nicht von SoX unterstützt werden.

3.3.3 Format von ASCII-Dateien

Das ASCII-Format ist recht nützlich für wissenschaftliche Zwecke und zu Ausbildungs-Zwecken. Aufgrund seines einfachen Aufbaus können Sie entweder eigene Dateien mit einem Texteditor schreiben oder auch die Ausgabe einer anderen Applikation verwenden und in ASCII konvertieren. Nachdem das Format *sehr* simpel ist sollten Sie keine grösseren Probleme haben falls notwendig einen Konverter zu schreiben; die meisten wissenschaftlichen Anwendungen haben eine Art eigenes ASCII-Format für den Export von Daten.

ASCII-Dateien haben einen relativ unkomplizierten Aufbau, nach folgenden Regeln:

1. Am Anfang der Datei kommt ein Block mit Datei-Eigenschaften, mit einer Eigenschaft pro Zeile.
2. Jede Eigenschafts-Zeile beginnt mit #.
3. Nach den Datei-Eigenschaften kommt eine Liste von Samples, mit einem Sample pro Zeile. Bei Verwendung von mehreren Spuren werden die Samples durch Komma getrennt.
4. Zeilen dürfen mit einem Zeilenvorschub und/oder einem Wagenrücklauf enden (so dass DOS-Dateien ebenfalls unterstützt werden). Beim Speichern von Dateien wird als Zeilende immer das Zeilenvorschub-Zeichen verwendet.
5. Leere Zeilen und Zeichen nach einem # gelten als Kommentare und werden ignoriert.
6. Werte müssen als vorzeichenbehafteten Integer-Format mit einem 24-Bit Wertebereich angegeben werden, dem internen Speicherformat von Kwave.
7. Alles nach einem # (mit Ausnahme der oben beschriebenen Dateieigenschaften) wird als Kommentar behandelt und wird ignoriert. Leere Zeilen werden ebenfalls ignoriert.

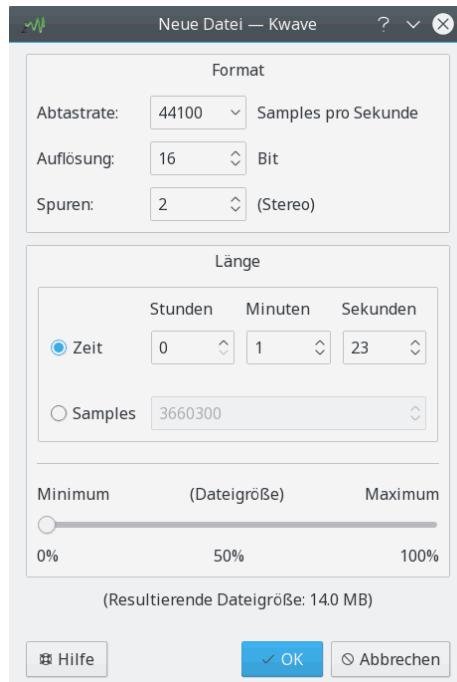
Hier ein Beispiel für eine einfache ASCII-Datei die einen Sinus mit elf Samples enthält:

Example 3.1 Inhalt einer ASCII-Datei mit elf Samples

```
## 'rate'=44100
## 'tracks'=2
## 'bits'=16
## 'length'=11
## 'Date'='2013-11-09'
## 'Software'='Kwave-0.8.11 für KDE 4.11.3'
5930496,      5930496 #          0
      0,     8388352 #          1
-5930752,      5930496 #          2
-8388608,      0 #          3
-5930752,     -5930752 #          4
      0,    -8388608 #          5
5930496,     -5930752 #          6
8388352,      0 #          7
5930496,      5930496 #          8
      0,     8388352 #          9
-5930752,      5930496 #         10
# EOF
```

3.4 Erstellen einer neuen Datei

Sie können eine neue leere Datei erstellen unter **Datei → Neu....**



Sie können eine Abtastrate, Auflösung in Bits pro Sample und die Anzahl der Spuren einstellen. Standardmäßig wird als Dateiformat ".wav" angenommen, was aber später beim Speichern der Datei geändert werden kann.

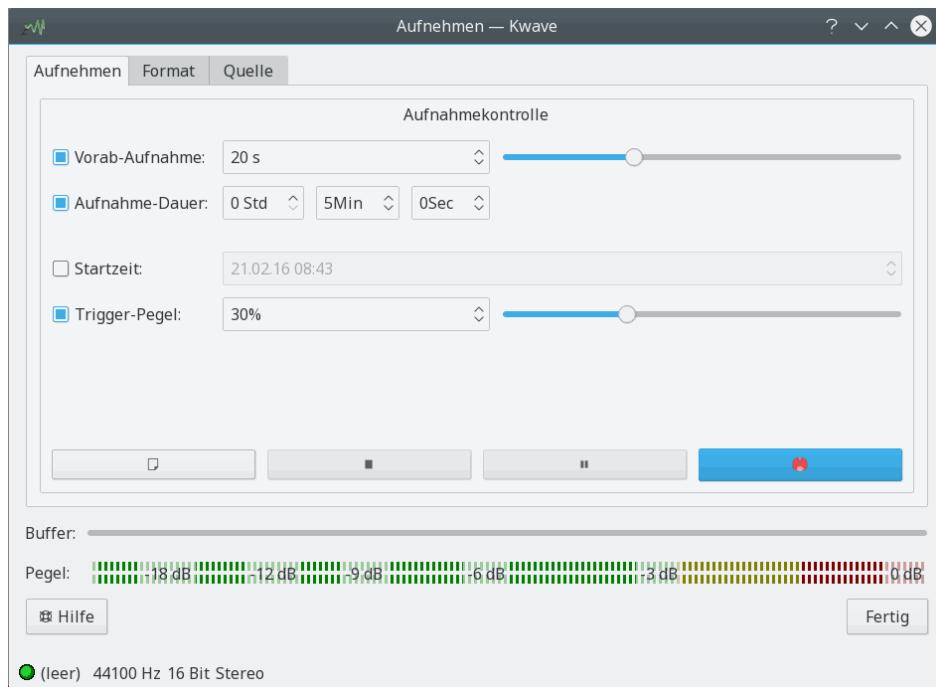
Die Länge einer neuen Datei kann über die Zeit (Stunden, Minuten, Sekunden) oder über die Anzahl der Samples eingestellt werden. Zusätzlich kann man sie relativ zur grössten möglichen Länge einstellen, die durch den vorhandenen Speicher und durch Kwave selbst (auf 2GB) begrenzt wird.

3.5 Aufnahme

Kwave ist in der Lage, von verschiedenen Quellen aufzunehmen, mit allen Abtastraten, Sample-Formaten und Einstellungen die Ihre Soundkarte unterstützt. Derzeit verwendet Kwave für die Aufnahme das alte OSS-Interface, und seit v0.7.4 auch das neuere und leistungsfähigere ALSA-System, dem System der Wahl für den Linux-Kernel 2.6.

Die Aufnahme kann im Menü unter **Datei → Aufnehmen** erreicht werden.

Hier ist ein Bildschirmfoto der ersten Seite des Kwave's Aufnahme-Dialogs mit den Aufnahmekontrollen, während einer laufenden Aufnahme. Wie in den meisten Dialogen von Kwave erhalten Sie einige Hilfestellungen und Kurzinfos von den Kontrollelementen.



Hier haben Sie folgende Einstellmöglichkeiten:

- **Vorab-Aufnahme:** Wenn Sie die Vorab-Aufnahme aktivieren, und die Aufnahme gestartet wird, nimmt Kwave in einen internen Ringpuffer auf, der einige Sekunden lang ist. Wenn Sie den Aufnahme ()-Knopf nochmals drücken, startet die Aufnahme wirklich, wobei die bereits im Ringpuffer aufgenommenen Daten erhalten bleiben. Dies ist beispielsweise nützlich, wenn Sie Ihr Lieblingslied vom Radio aufnehmen wollen, aber zu spät mitkriegen dass das Stück bereits gestartet ist. In diesem Fall können Sie immer noch den Aufnahme-Knopf drücken und den Start des Stücks aus dem beziehen was Kwave bereits aufgenommen hat. Somit verpassen Sie nie mehr einen Anfang.
- **Aufnahmedauer:** Wenn die Länge der Aufnahme auf eine bestimmte Zeit begrenzt werden soll, können Sie diese Einstellung aktivieren und eine Zeit in Stunden, Minuten, Sekunden einstellen. Wenn Sie diese Einstellung nicht aktivieren, läuft die Aufnahme so lange bis Sie den Stop ()-Knopf drücken.
- **Startzeit:** Wenn diese Einstellung aktiviert ist, können Sie ein Datum und eine Uhrzeit angeben zu der die Aufnahme gestartet wird. Bitte beachten Sie dass die Aufnahme sofort beginnt wenn die eingestellte Zeit in der Vergangenheit liegt.

- **Trigger-Pegel:** Wenn aktiviert, startet die Aufnahme erst wenn die Lautstärke des Eingangssignals einen bestimmten Pegel überschreitet, der zwischen 0 und 100% der höchsten möglichen Eingangs-Lautstärke. Dies ist nützlich wenn Sie nicht die Stille am Anfang aufgenommen haben wollen.



- Der **Neu** ()-Knopf ist aktiv wenn keine Aufnahme läuft oder die Aufnahme beendet ist. Der Inhalt der aktuellen Datei wird verworfen.



- Der **Stop** ()-Knopf ist aktiv während die Aufnahme oder Vorab-Aufnahme läuft oder während Kwave auf das Erreichen eines Trigger-Pegels wartet. Beim Klicken wird der aktuelle Vorgang angehalten.



- Der **Pause** ()-Knopf ist aktiv während die Aufnahme oder Vorab-Aufnahme läuft. Beim ersten Klicken wird die Aufnahme angehalten und der Knopf fängt an zu blinken. Beim zweiten Klicken hört der Knopf auf zu blinken und die Aufnahme geht sofort weiter, ohne auf das Erreichen eines Trigger-Pegels zu warten.



- The **Aufnahme** () startet die Aufnahme und Vorab-Aufnahme, abhängig von den oben aktivierten Einstellungen.

1. Wenn weder Vorab-Aufnahme noch Trigger-Pegel aktiviert wurden, startet die Aufnahme sofort beim Klicken des Aufnahmebuttons.
2. Wenn die Vorab-Aufnahme aktiviert wurde und ein Trigger-Pegel gesetzt wurde, wartet Kwave nach dem ersten Klicken auf das Erreichen des Trigger-Pegels. Während dem Warten können Sie den Start der Aufnahme erzwingen indem Sie den Aufnahmebutton nochmals klicken, ansonsten startet die Aufnahme sobald der Trigger-Pegel erreicht wurde.
3. Wenn die Vorab-Aufnahme aktiviert wird, startet das erste Klicken nur die Vorab-Aufnahme, erst beim zweiten Klick wird die Aufnahme wirklich gestartet.

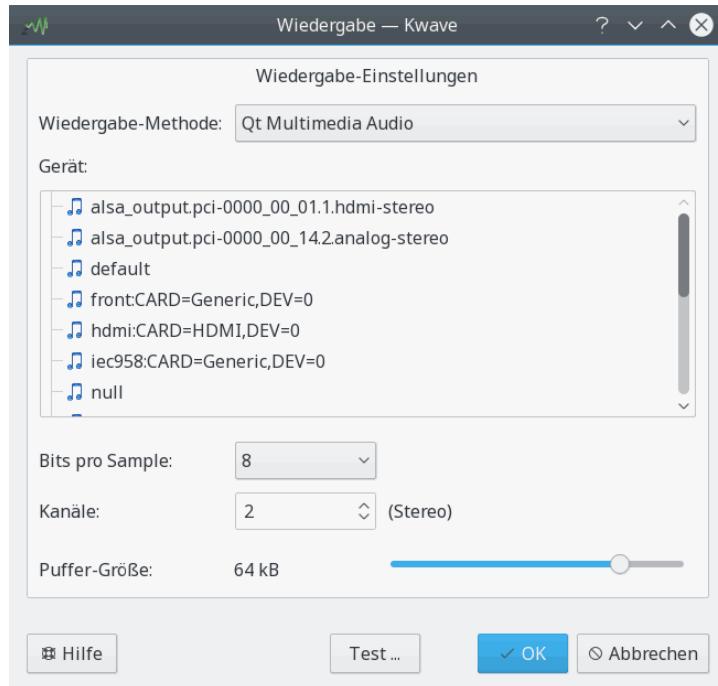
3.6 Wiedergabe

Je nachdem welche Optionen zur Kompilierzeit gesetzt waren, kann Kwave mit Hilfe folgender Wiedergabe-Methoden abspielen:

- **ALSA (Advanced Linux Sound Architecture):** Löst OSS ab, unterstützt mehr Features und mehr Hardware als OSS. Kann ebenso wie OSS mit anderen Sound-Applikationen kollidieren, hat aber ein Modul namens „DMIX“ als Ausweg. Diese Wiedergabe-Methode sollte für Sie die beste Wahl sein!
- **OSS (Open Sound System):** Die älteste Linux-Implementierung, Wiedergabe in Mono und Stereo. Veraltet seit Linux-Kernel 2.6, aber immer noch weit verbreitet. Kann mit anderen Sound-Applikationen kollidieren, es können nicht mehrere Applikationen gleichzeitig OSS-Wiedergabe nutzen !

Bevor Sie erstmals Dateien abspielen, sollten Sie den Dialog der Wiedergabe-Einstellungen ansehen:

Das Kwave Handbuch



Momentan unterstützt Kwave die Wiedergabe mit 8 und 16 Bits, mit Mono und Stereo via OSS, sowie alle andern Modi die von Ihrer Sound-Hardware unterstützt werden mit Hilfe des ALSA-Systems.

Wenn Ihre Datei mehr Spuren besitzt als Ihre Wiedergabe-Hardware erlaubt, werden alle Kanäle zur Wiedergabe zusammengemischt. Wenn Sie beispielsweise eine Datei mit drei Spuren abspielen wollen, so wird am linken Kanal die Spur 0 (oben) und die Hälfte von Spur 1 (mitte) abgespielt, am rechten Kanal die andere Hälfte von Spur 1 und der Spur 2 (unten).

Um eine saubere Wiedergabe ohne Unterbrechungen zu erreichen, sollten Sie die Puffer-Grösse auf einen sinnvollen Wert einstellen. Wenn Sie Probleme mit Unterbrechungen während der Wiedergabe haben, sollten Sie hier den Puffer vergrössern. Je grösser Sie jedoch den Puffer einstellen, desto grösser wird die Verzögerung zwischen dem hörbaren Sound und der angezeigten Wiedergabe-Position im Hauptfenster.

Der Dialog der Wiedergabe-Einstellungen enthält auch einen Knopf um einen einfachen Test-Ton abzuspielen. Sie sollten einen 440Hz-Ton hören, der der Reihe nach über alle Lautsprecher wandert.

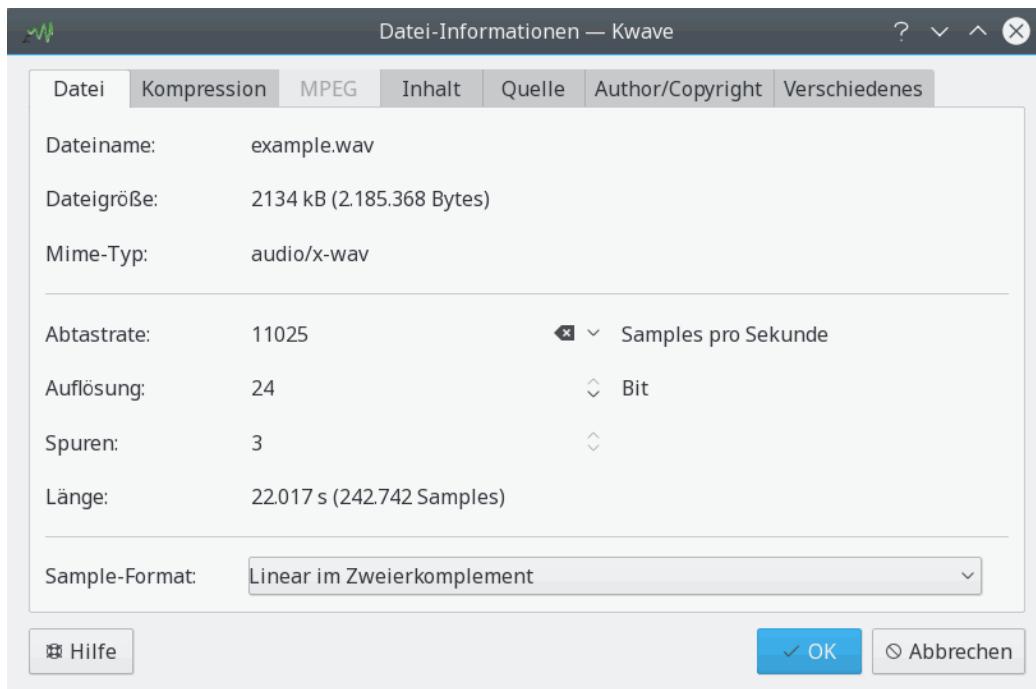
Sobald Sie die Wiedergabe konfiguriert haben, können Sie die Wiedergabe-Kontrollelemente des Kwave-Hauptfensters aus dem **Wiedergabe**-Menü oder mit Tastaturkürzel verwenden:

- **Wiedergabe → Start (P)**: Startet die Wiedergabe vom Anfang der aktuellen Auswahl an, oder von der aktuellen Cursorposition wenn nichts ausgewählt wurde. Es wird nur ein mal abgespielt.
- **Wiedergabe → Schleife**: wie zuvor, jedoch mit Wiederholung in einer Schleife.
- **Wiedergabe → Pause (Leertaste)**: Hält die Wiedergabe an der aktuellen Position an. Nur verfügbar während die Wiedergabe läuft.
- **Wiedergabe → Fortsetzen (Leertaste)**: Setzt die Wiedergabe an der Stelle fort an der sie angehalten wurde. Nur verfügbar wenn die Wiedergabe angehalten wurde.
- **Wiedergabe → Stop (Esc)**: Beendet die Wiedergabe, geht zurück an den Start der Auswahl.

3.7 Datei-Eigenschaften

Kwave kann verschiedene Zusatzinformationen verwalten, die mit in einer Audio-Datei gespeichert wurden. Es versucht, so viele dieser Informationen wie möglich zu importieren und zu exportieren. Wenn Sie zum Beispiel MP3-Dateien mit ID3-Tags importieren, bleiben diese beim Speichern als Wave-Datei nicht erhalten. Wenn Kwave Zusatzinformationen beim Speichern verlieren würde, erscheint eine Warnung.

Datei-Informationen können Sie ansehen und bearbeiten unter **Bearbeiten → Datei-Informationen...**. Hier können auch Sie Dinge wie das Sample-Format, Auflösung, und Kompression einstellen.



3.8 Zoomen und Navigieren

Kwave ermöglicht es auf verschiedene Arten und Weisen zu zoomen und zu navigieren: über Tastaturkürzel, Werkzeugeiste und mit der Maus. Die folgenden Abschnitte geben eine Übersicht wie diese Funktionen verwendet werden.

3.8.1 Hinein- und Heraus-zoomen

- **Vergrößern:** vergrößert die Ansicht um mehr Details darzustellen, vergrößert um Faktor 3.
 - Menüeintrag / Tastenkürzel: **Ansicht → Vergrößern (Strg+)**
 - Werkzeugeisten-Schaltfläche: „Vergrößern“
- **Verkleinern:** verkleinert die aktuelle Ansicht um weniger Details darzustellen, verkleinert um Faktor 3.
 - Menüeintrag / Tastenkürzel: **Ansicht → Verkleinern (Strg--)**
 - Werkzeugeisten-Schaltfläche: „Verkleinern“
- **Zoom auf Auswahl:** wählt den Vergrößerungsfaktor so dass die aktuelle Auswahl vollständig ins die aktuelle Ansicht eingepasst wird.

- Menü-Eintrag / Tastenkürzel: **Ansicht → Zoom auf Auswahl (Strg-Leertaste)**
- Werkzeugeisten-Schaltfläche: „Vergrößern auf Auswahl“
- *Auf das ganze Signal zoomen*: Wählt einen Vergrößerungsfaktor der das gesamte Signal in der aktuellen Ansicht sichtbar macht.
 - Menü-Eintrag: **Ansicht → Zoom auf das gesamte Signal**
 - Werkzeugeisten-Schaltfläche: „auf alles vergrößern“
- *Auf 100% vergrößern*: vergrößert auf einen Massstab bei dem ein Sample einem Pixel auf dem Bildschirm entspricht.
 - Menü-Eintrag: **Ansicht → auf 100% vergrößern**
 - Werkzeugeisten-Schaltfläche: „auf 100% vergrößern“
- *Vordefinierter Faktor*: Verwendet einen Vergrößerung-Faktor aus der Auswahlbox in der Werkzeugeiste.

3.8.2 Ansicht nach links und rechts verschieben

- *Gehe zu Position*: Öffnet einen Dialog um eine Position auszuwählen zu der die aktuelle Ansicht verschoben wird.
Menüeintrag / Tastenkürzel: **Ansicht → Gehe zu Position... (Strg-G)**
- *zum Anfang*: verschiebt den sichtbaren Bereich so dass er am Anfang des Signals beginnt.
Menüeintrag / Tastenkürzel: **Ansicht → Anfang (Strg-Pos 1)**
- *ans Ende*: verschiebt die aktuelle Ansicht so dass sie beim Ende des Signals endet.
Menüeintrag / Tastenkürzel: **Ansicht → Ende (StrgEnde)**
- *vorherige Seite*: verschiebt den sichtbaren Bereich zu einer Position direkt vor der aktuellen Ansicht (nach links).
Menüeintrag / Tastenkürzel: **Ansicht → Vorherige Seite (Bild auf)**
Werkzeugeisten-Schaltfläche: „vorherige Seite“
- *Nächste Seite*: verschiebt den sichtbaren Bereich auf eine Position direkt nach der aktuellen Ansicht (nach rechts).
Menüeintrag / Tastenkürzel: **Ansicht → Nächste Seite (Bild ab)**
Werkzeugeisten-Schaltfläche: „nächste Seite“
- *Ansicht nach rechts verschieben*: verschiebt die Ansicht um 1/3 des aktuell sichtbaren Bereichs nach rechts.
Menü-Eintrag / Tastenkürzel: **Ansicht → nach rechts verschieben (Pfeil rechts)**
Werkzeugeisten-Schaltfläche: „nach rechts scrollen“
- *Ansicht nach links verschieben*: verschiebt die Ansicht um 1/3 des aktuell sichtbaren Bereichs nach links.
Menü-Eintrag / Tastenkürzel: **Ansicht → nach links verschieben (Pfeil links)**
Werkzeugeisten-Schaltfläche: „nach links scrollen“

3.8.3 Verwenden der Übersicht

Das Hauptfenster von Kwave enthält oberhalb des horizontalen Scrollbalkens eine kleine *Übersicht* über das gesamte Signal. Diese Übersicht stellt ebenfalls Funktionen zum Navigieren zur Verfügung:

- *einfacher Klick mit der linken Maustaste*: verschiebt die aktuelle Ansicht direkt zur angeklickten Position.
- *Doppelklick mit der linken Maustaste*: verschiebt die aktuelle Ansicht direkt zur angeklickten Position und vergrößert zusätzlich.
- *Doppelklick mit der linken Maustaste mit gedrückter Umschalt-Taste*: verschiebt die aktuelle Ansicht direkt zur angeklickten Position und verkleinert zusätzlich.

3.8.4 Vertikale Vergrößerung

Sie können die aktuelle Ansicht vertikal vergrößern und verkleinern indem Sie die **Alt-Taste** drücken während Sie mit dem Mausrad rollen.

3.9 Die Funktionsweise der Auswahl

Kwave erlaubt es Ihnen, einen kontinuierlichen Bereich von Samples wie auch jede Kombination von Spuren (wenn Sie eine Mehrspur-Datei bearbeiten) auszuwählen. Wenn Sie einen Sample-Bereich (Zeitrahmen) ausgewählt haben, werden alle folgenden Befehle darauf angewendet, wobei das Deselektieren einer Spur den Inhalt nicht verändert.

3.9.1 Auswahl von Samples

Das Auswählen oder das Deselektieren einer Spur ist recht einfach. Klicken Sie einfach auf das Lampensymbol auf der linken Seite des Signalfensters um seinen Status zu verändern:

-  ein grünes Licht bedeutet "aktiviert", wohingegen
-  eine rote Lampe bedeutet "deaktiviert".

ANMERKUNG

Hinweis: Wird eine Spur deseletiert, wird sie bei der Wiedergabe nicht hörbar sein!

3.9.2 Auswahl von Samples

Wenn Sie einen Sample-Bereich in Kwave auswählen, wird dieser Bereich *inclusive* sein. Das heisst, dass das erste und das letzte ausgewählte Sample mit zur Auswahl gehören und benutzt werden für die folgenden Aktionen. Deshalb wird, selbst wenn Sie keinen *Bereich* sondern nur ein einzelnes Sample ausgewählt haben, die Auswahl niemals wirklich "leer" sein. So zum Beispiel wird, selbst wenn Sie keinen Bereich ausgewählt haben, die "Löschen" Funktion auf dieses einzelne Sample angewendet.

Der einfachste Weg einen Sample-Bereich auszuwählen ist einfach mit der Maus. Es funktioniert so, wie Sie es von anderen Anwendungen her bereits gewohnt sind: drücken Sie einfach die linke Maustaste auf den Punkt, beim dem die Auswahl beginnen soll und lassen die Maustaste am Endpunkt wieder los.

Wenn Sie die Auswahl verschieben oder ihren Start- oder Endpunkt anpassen wollen, können Sie den Mauscursor an den Rand der Auswahl bewegen bis sich der Standard-Pfeil Cursor in einen Links-Rechts Pfeil Cursor geändert hat, um dann die linke Maustaste zu drücken und die Anpassung vorzunehmen.

Sie können ebenso die Auswahl auf einen bestimmten Punkt erweitern oder verkürzen indem Sie die **Umschalt-Taste** gedrückt halten und mit der linken Maustaste klicken. Abhängig davon welcher Rand näher ist, wird der linke oder rechte Rand der Auswahl neu gesetzt. Falls nichts ausgewählt ist, wird die Auswahl vom Anfang gesetzt.

Es sind auch einige Funktionen über das Menü und über einige Tastatur- Kurztasten (Shortcuts) verfügbar:

- Auswahl des gesamten Signals: **Bearbeiten** → **Auswahl** → **Alles (Strg-A)**
- Bereich auswählen: **Bearbeiten** → **Auswahl** → **Bereich (R)**
- die aktuell sichtbare Fläche: **Bearbeiten** → **Auswahl** → **Sichtbarer Bereich (V)**
- den nächsten Sample-Block, beginnend ein Sample nach dem Ende der aktuellen Auswahl und mit der gleichen Länge: **Bearbeiten** → **Auswahl** → **Nächste (Umschalt-+)**
(Hinweis: benutzen Sie die "+" Taste vom numerischen Ziffernblock!)
- den vorhergehenden Sample-Block, bis ein Sample vor dem Anfang der aktuellen Auswahl und mit der gleichen Länge: **Bearbeiten** → **Auswahl** → **Vorherige (Umschalt--)**
(Hinweis: benutzen Sie die "+" Taste vom numerischen Ziffernblock!)
- entfernen der Auswahl und markieren von "nichts": **Bearbeiten** → **Auswahl** → **Nichts (N)**
- erweitern der Auswahl bis zum Anfang des Signals (erstes Sample): **Bearbeiten** → **Auswahl** → **bis zum Anfang (Umschalt-Pos 1)**
- erweitern der Auswahl bis zum Ende des Signals (letztes Sample): **Bearbeiten** → **Auswahl** → **bis zum Ende (Umschalt-Ende)**
- Erweitern der Auswahl links und rechts bis zum nächsten Markierung (oder Start / Ende des Signals falls keiner vorhanden), ausgehend von der aktuellen Cursor-Position: **Bearbeiten** → **Auswahl** → **Auf Markierungen erweitern (E)**
- Auswahl des Bereiches zwischen den nächsten beiden Markierungen, die rechts von der aktuellen Auswahl liegen oder bis zum Ende des Signals: **Bearbeiten** → **Auswahl** → **Zu den nächsten Markierungen (Strg-Umschalt-N)**
- Auswahl des Bereiches zwischen den vorhergehenden beiden Markierungen, die links von der aktuellen Auswahl liegen oder bis zum Anfang des Signals: **Bearbeiten** → **Auswahl** → **Zu den vorhergehenden Markierungen (Strg-Umschalt-P)**

3.10 Zwischenablage

Kwave verwendet die Zwischenablage von Plasma. Damit ist es möglich Audio-Daten zwischen verschiedenen Kwave-Fenstern auszutauschen. Außerdem lassen sich möglicherweise Daten zwischen Kwave und anderen Audio-Programmen austauschen, abhängig von deren Fähigkeit die Plasma-Zwischenablage zu nutzen.

Beim kopieren in die Zwischenablage über die Kopieren-Funktion verwendet Kwave den Medientyp audio/vnd.wave, gemäss [RFC 2361](#), wie in dem bekannten wav-Format. Beim Einfügen von der Zwischenablage in Kwave werden alle Datenformate unterstützt die als Datei-Import-Formate verfügbar sind, wie beispielsweise Ogg/Vorbis, FLAC und andere.

3.11 Drag and Drop

Kwave unterstützt das KDE Frameworks Drag and Drop-Protokoll. Dies ermöglicht Ihnen, Dateien einfach mittels der Maus in einem Dolphin- oder Konqueror-Fenster oder vom Desktop aufzunehmen und sie über einem Fenster von Kwave fallen zu lassen, um sie zu öffnen.

Bitte beachten Sie, dass wenn eine Datei über einen Kwave Fenster fallen gelassen wird, welches bereits eine geöffnete Datei enthält, diese zuerst geschlossen wird und dann erst die losgelassene Datei geöffnet wird. Wenn Sie dies nicht wollen, sollten Sie zuerst ein neues leeres Kwave Fenster öffnen.

Sie können auch einen Bereich von Samples auswählen und per Drag und Drop in ein anderes Kwave-Fenster verschieben. Standardmäßig wird der Vorgang im *Verschieben*-Modus durchgeführt, wobei der ausgewählte Bereich von seiner ursprünglichen Stelle gelöscht und an der Position des Fallenlassens eingefügt wird. Durch Drücken der **Strg**-Taste können Sie dieses Verhalten ändern und stattdessen den *Kopieren*-Modus verwenden.

Kapitel 4

Automatisieren und Skripten mit Kwave

Kwave verwendet seit der ersten Version intern Text-Befehle. Diese werden verwendet um das Menü aufzubauen, die graphische Oberfläche zu steuern, eingebaute Effekte und Erweiterungen zu steuern. Die einzelnen Befehle werden später im Kapitel beschrieben.

4.1 Allgemeine Syntax

- Alle Befehle bestehen aus einem *Befehls-Namen* und einer optionalen *Parameterliste* in runden Klammern, je nach Befehl.
- Zulässige Zeichen für Befehls-Namen sind *Buchstaben*, *Ziffern* und *Doppelpunkt*. Groß-/Kleinschreibung wird unterschieden, alle Befehle sind in Kleinschreibung.
- Parameter innerhalb einer Parameterliste werden durch *Komma* getrennt.
- Numerische Parameter können als Festkomma- oder als Gleitkomma-Zahlen angegeben werden, *als Dezimaltrennzeichen wird ein Punkt verwendet*.
- Text-Parameter werden automatisch zugeschnitten (alle Leerzeichen am Anfang und am Ende werden entfernt). Sollte das nicht gewünscht sein, so kann der Parameter in Anführungszeichen (") gesetzt werden. Falls ein Text Sonderzeichen enthält (wie beispielsweise „“, „;“, „#“ or a „\“), so muss diesen ein Escape-Zeichen „\“ vorangestellt werden .
- Mehrere Befehle können mit einem „;“ als Trennzeichen zu einer Befehlsliste zusammengefügt werden.

Beispiel:

```
fileinfo(Comments,"Dies ist ein \"Beispiel\"-Kommentar.")
```

Dieses Beispiel besteht aus dem Befehl **fileinfo** und hat zwei Parameters: das Schlüsselwort *Comments* und den Text „*Dies ist ein "Beispiel"-Kommentar.*“. (Diese Parameter sind im zugehörigen Abschnitt der Befehlsreferenz beschrieben).

4.2 Verwenden der Befehlszeile

Zusätzlich zu den im Abschnitt [Befehlszeile](#) beschriebenen Parametern zum Starten von Kwave im minimierten Zustand oder ohne Begrüßungsbildschirm können Sie auch Text-Befehle per Befehlszeile übergeben, kodiert als spezielle Adresse (URI):

```
kwave:Befehl[?parameter[,Parameter ...] ]
```

Die Regeln zur Übersetzung eines Kwave-Text-Befehls in eine gültige Adresse (URI) sind wie folgt:

- Die Adresse (URI) startet mit dem Wort `kwave`, gefolgt von einem „:“ und dem Befehls-Namen.
- Falls der Befehl Parameter besitzt, müssen diese nach dem Befehl, getrennt mit einem „?“ angehängt werden.
- Mehrere Parameter können mit einem „,“ als Trennzeichen angehängt werden.
- Alle Sonderzeichen in Befehlsnamen und Parameter müssen wie eine URL kodiert werden. Hier eine Liste von Zeichenersetzungen:

Orig-inal	kodi-ert		Orig-inal	kodi-ert		Orig-inal	kodi-ert		Orig-inal	kodi-ert
(Lee-rzeic-hen)	%20		(%28		:	%3A		\	%5C
!	%21)	%29		;	%3B]	%5D
”	%22		*	%2A		<	%3C		^	%5E
#	%23		+	%2B		=	%3D		-	%5F
\$	%24		,	%2C		>	%3E		'	%60
%	%25		-	%2D		?	%3F		{	%7B
&	%26		.	%3E		@	%40			%7C
'	%27		/	%2F		[%5B		}	%7D
									~	%7E

Tabelle 4.1: Übersetzungstabelle für URL-Kodierung

4.3 Kwave-Skripte

4.3.1 Allgemeine Struktur

Ein Kwave-*Skript* besteht aus einer Liste von Zeilen, wobei jede Zeile in eine der folgenden Kategorien eingeteilt werden kann:

- ein einzelner *Befehl*,
- eine *Befehls-Liste*, mit zwei oder mehr per „,“ voneinander getrennten Befehlen.
- ein *Kommentar*,
- eine *Markierung*
- oder eine *Leerzeile*, die nur aus Leerzeichen besteht

4.3.2 Kommentare und Leerzeilen

Alle Zeichen die einem „#“ folgen (außer wenn in Anführungszeichen gesetzt oder mit vorgestelltem Escape-Zeichen) werden als Kommentare behandelt, sie werden stillschweigend ignoriert.

Zeilen die nur aus Leerzeichen oder Kommentaren bestehen werden ebenfalls ignoriert.

4.3.3 Beenden

Ein Kwave-Skript beendet sich entweder wenn alle Befehle ohne einen Fehler ausgeführt wurden oder sobald ein Befehl einen Fehlercode zurück gibt. Es gibt keinen speziellen Befehl zum Abbrechen eines Skriptes. Falls Sie jedoch eine Möglichkeit vorsehen wollen dass der Anwender ein Skript beenden kann, so steht Ihnen der Befehl „**msgbox**(Text)¹“ zur Verfügung. Dieser zeigt ein Benachrichtigungs-Fenster mit den beiden Schaltflächen **OK** (um das Skript weiter auszuführen) und der Schaltfläche **Abbrechen** (um das Skript zu beenden).

4.3.4 Markierungen

Zeilen die nur aus einem Bezeichner gefolgt von einem „::“ bestehen werden als *Markierungen* betrachtet. Auf diese kann man sich an anderen Stellen im Skript mit dem Schlüsselwort **GOTO**¹ beziehen, das bewirkt dass die Ausführung des Skriptes an der Stelle mit der angegebenen Markierung fortgesetzt wird (siehe Beispiel unten).

Eine Zeile mit einer Markierung darf nach dem „::“ keinen anderen Inhalt (mit Ausnahme von Kommentaren und Leerzeichen) enthalten.

Beispiel:

```
start: # <= dies ist eine Markierung
        # tue irgendwas...
        msgbox(noch einmal?)
        GOTO start
```

4.4 Befehlsreferenz

4.5 a

4.5.1 **about_kde**

4.5.1.1 Syntax: **about_kde()**

Zeigt ein Dialogfenster mit Informationen über die KDE-Gemeinschaft.

4.5.2 **add_track**

4.5.2.1 Syntax: **add_track()**

Fügt eine neue Spur nach allen vorhandenen Spuren an.

¹Hinweis: Bitte verwechseln Sie nicht das Schlüsselwort „GOTO“ mit dem Textbefehl „**goto** (Position)“ !

4.5.2.2 Siehe auch

[insert_track\(\)](#)

4.6 c

4.6.1 clipboard_flush

4.6.1.1 Syntax: clipboard_flush()

Verwirft den aktuellen Inhalt der Zwischenablage (kann etwas Arbeitsspeicher frei machen).

4.6.2 close

4.6.2.1 Syntax: close()

Schließt die aktuelle Datei. Wenn die graphische Oberfläche im MDI- oder Unterfenster-Modus konfiguriert ist, schließt dies auch das zugehörige Unterfenster.

4.6.2.2 Siehe auch

[open\(*Dateiname*\), quit\(\)](#)

4.6.3 continue

4.6.3.1 Syntax: continue()

Entspricht der Werkzeugleisten-Schaltfläche **Fortsetzen** und lässt die Wiedergabe fortsetzen wenn sie angehalten wurde.

4.6.3.2 Siehe auch

[pause\(\)](#)

4.6.4 copy

4.6.4.1 Syntax: copy()

Kopiert den Inhalt der aktuellen Auswahl in die Zwischenablage. Wenn die Auswahl leer ist, bewirkt dieser Befehl nichts und der Inhalt der Zwischenablage bleibt unverändert. Es wird nur der Inhalt der aktuell ausgewählten Spuren in die Zwischenablage kopiert.

4.6.4.2 Siehe auch

[paste\(\)](#)

4.6.5 crop

4.6.5.1 Syntax: `crop()`

Schneidet das Signal auf die aktuelle Auswahl zu indem alles was sich vor oder nach der Auswahl befindet gelöscht wird. Wirkt sich auf alle Spuren aus. Wenn kein Bereich markiert ist bewirkt dieser Befehl nichts.

4.6.6 cut

4.6.6.1 Syntax: `cut()`

Kopiert den Inhalt der aktuellen Auswahl in die Zwischenablage und löscht ihn aus dem Signal. Wenn die aktuelle Auswahl leer ist bewirkt dieser Befehl nichts und der Inhalt der Zwischenablage bleibt unverändert. Nur der Inhalt der ausgewählten Spuren wird in die Zwischenablage kopiert, der ausgewählte Bereich wird jedoch von allen Spuren gelöscht.

4.7 d

4.7.1 delayed

4.7.1.1 Syntax: `delayed(Millisekunden, Befehl)`

Führt einen *Befehl* nach einer angegebenen Verzögerung aus. Bitte beachten Sie dass der Befehl asynchron ausgeführt wird nachdem die angegebene Zeit abgelaufen ist. Mehrere Befehle können eingereiht werden, die Verzögerungszeit beginnt dann jeweils nach der Ausführung des letzten eingereihten Befehls. Dieser Befehl ist gedacht um Befehle vorzubereiten die beim Erstellen von BildschirmOTOS benötigt werden.

4.7.1.2 Parameter

<i>Millisekunden</i> :	Anzahl der ganzen Millisekunden die vor der Ausführung des Befehls gewartet werden sollen
<i>Befehl</i> :	Ein Befehl der nach der angegebenen Verzögerung ausgeführt werden soll, inklusive Parameter

4.7.1.3 Siehe auch

[sync\(\)](#), [window:resize\(\)](#), [window:click\(\)](#), [window:sendkey\(\)](#), [window:close\(\)](#), [window:screenshot\(\)](#)

4.7.2 delete

4.7.2.1 Syntax: `delete()`

Löscht den aktuell ausgewählten Bereich von Samples. Wenn die Auswahl leer ist bewirkt dieser Befehl nichts. Wirkt sich auf alle Spuren aus.

4.7.3 delete_track

4.7.3.1 Syntax: `delete_track(index)`

Löscht eine Spur anhand des Index (beginnend bei Null). Wenn keine Spur mit dem angegebenen Index existiert, kehrt dieser Befehl mit einem Fehler zurück.

4.7.3.2 Parameter

<i>Index:</i>	Index der zu löschen Spur, beginnend bei 0
---------------	--

4.7.4 dump_metadata

4.7.4.1 Syntax: `dump_metadata()`

Gibt eine Liste von Metadaten auf der Konsole aus, zu diagnostischen Zwecken. (Nur verfügbar wenn Kwave mit der aktivierten Option `WITH_DEBUG` generiert wurde).

4.8 e

4.8.1 expandtolabel

4.8.1.1 Syntax: `expandtolabel()`

Erweitert die Auswahl auf die Markierungen links und rechts von der aktuellen Auswahl. Wenn eine Grenze der Auswahl bereits auf einer Markierung liegt, wird sie nicht verändert. Wenn links oder rechts von der Auswahl keine Markierung liegt, wird sie zum Start oder Ende der Datei erweitert.

4.9 f

4.9.1 fileinfo

4.9.1.1 Syntax: `fileinfo(index)`

Setzt einen Eintrag der Dateiinformationen auf einen neuen Wert.

4.9.1.2 Parameter

<i>Schlüsselwort:</i>	Schlüsselwort des Eintrags
<i>Wert:</i>	Wert des Eintrags

4.9.2 forward

4.9.2.1 Syntax: `forward()`

Entspricht der Werkzeugleisten-Schaltfläche **Vorspulen**. Wenn die Wiedergabe aktuell läuft, wird um 1/10 des sichtbaren Bereichs vor gespult. Wenn die Wiedergabe nicht läuft, entspricht dieser Befehl dem Befehl `view:scroll_right()`.

4.9.2.2 Siehe auch

[view:scroll_right\(\)](#), [rewind\(\)](#)

4.10 g

4.10.1 goto

4.10.1.1 Syntax: `goto(position)`

Setzt den Cursor an die angegebene Position und verschiebt die aktuelle Ansicht so daß die Position angezeigt wird. Anschließend hat die Auswahl die Länge Null.

4.10.1.2 Parameter

Pos:

Position in Samples an die gesprungen werden soll

4.11 i

4.11.1 insert_at

4.11.1.1 Syntax: `insert_at(position)`

Fügt den Inhalt der Zwischenablage an der angegeben Position ein, so wie der Befehl `paste()`. Ist die Zwischenablage leer bewirkt dieser Befehl nichts.

4.11.1.2 Parameter

Pos:

Position in Samples an der eingefügt werden soll

4.11.1.3 Siehe auch

[paste\(\)](#)

4.11.2 insert_track

4.11.2.1 Syntax: `insert_track(Index)`

Fügt an dem angegebenen Index eine neue Spur hinzu, mit der Länge und Abtastrate des aktuellen Signals. Wenn der Index größer oder gleich der aktuellen Anzahl der Spuren ist, wird die Spur am Ende angehängt, so wie durch den Befehl `add_track()`. Der Index aller Spuren ab der Einfüge-Position wird hierdurch um eins erhöht.

4.11.2.2 Parameter

<i>Index:</i>	Index der einzufügenden Spur, beginnend bei 0
---------------	---

4.11.2.3 Siehe auch

[add_track\(\)](#)

4.12 l

4.12.1 label:add

4.12.1.1 Syntax: `label:add(pos[, text])`

Fügt eine Markierung an der angegebenen Position hinzu. Falls an der angegeben Position bereits eine Markierung existiert, bewirkt dieser Befehl nichts. Der Markierung kann auch eine Beschreibung zugewiesen werden.

4.12.1.2 Parameter

<i>Pos:</i>	Position in Samples an der die Markierung hinzugefügt werden soll
<i>Text:</i>	eine kurze Beschreibung (optional)

4.12.2 label:delete

4.12.2.1 Syntax: `label:delete(Index)`

Löscht eine Markierung anhand des Index (beginnend bei Null), oder alle Markierungen wenn als Index der Spezialwert -1 verwendet wird. Wenn an dem angegeben Index keine Markierung existiert, bewirkt dieser Befehl nichts.

4.12.2.2 Parameter

Index:

Index der zu löschen Markierung,
beginnend bei 0, oder -1 um alle
Markierungen zu löschen

4.12.3 label:edit



4.12.3.1 Syntax: label:edit(*Index*)

Öffnet ein Dialogfenster, mit dem der Anwender die Position und die Beschreibung einer Markierung anhand ihres Null-basierten Index ändern kann. Wenn keine Markierung mit dem angegeben Index existiert, bewirkt diese Funktion nichts.

4.12.3.2 Parameter

Index:

Index der zu bearbeitenden Markierung,
beginnend bei 0

4.12.4 loadbatch

4.12.4.1 Syntax: loadbatch(*Dateiname*)

Öffnet eine Kwave-Skript-Datei und führt die darin enthaltenen Befehle aus. Verwendet den Kontext der aktuell geöffneten Datei oder des aktuell geöffneten Fensters falls keine Datei geladen ist.

4.12.4.2 Parameter

<i>Dateiname:</i>	Name der Kwave-Skripts-Datei, inklusive Pfad und Dateierweiterung
-------------------	---

4.12.5 loop

4.12.5.1 Syntax: `loop()`

Entspricht der Werkzeugleisten-Schaltfläche **Schleife**. Startet die Wiedergabe (falls sie noch nicht läuft) und lässt sie in einer Schleife spielen.

4.12.5.2 Siehe auch

[playback_start\(\)](#)

4.13 m

4.13.1 menu

4.13.1.1 Syntax: `menu (Befehl, Pfad, [Tastenkürzel], [ID])`

Dies ist ein sehr mächtiger Befehl, der verwendet wird um Menüeinträge hinzuzufügen oder zu ändern. Man legt damit fest welcher *Befehl* ausgeführt wird wenn der Eintrag aktiviert wird, welches *Icon* im Menü angezeigt und welches *Tastenkürzel* verwendet wird. Jeder Menüeintrag kann *deaktiviert* oder *ausgeblendet* werden, ihm kann eine eindeutige *ID* zugewiesen werden und er kann auch zu einer oder mehreren *Menü-Gruppen* hinzugefügt werden.

Normalerweise brauchen Sie dies nicht in einem Kwave-Skript!

4.13.1.2 Parameter

<i>Befehl:</i>	Ein Befehl (inklusive Parameter) oder eine Befehlsliste die ausgeführt wird wenn der Menüeintrag aktiviert wird. Wenn der Menüeintrag keinen zugehörigen Befehl besitzt (beispielsweise bei wenn es sich um ein Untermenü handelt und nicht um einen Menüeintrag), verwenden Sie bitte den speziellen Befehl <code>ignore()</code> .
----------------	--

Das Kwave Handbuch

<i>Pfad:</i>	Der Pfad innerhalb des Menüs, mit einem „/“ als Trennzeichen. Der letzte Teil des Pfads kann ein <i>Unterbefehl</i> sein, der eine Eigenschaft des Eintrags verändern kann (siehe unten). Der letzte Teil des Pfads (der kein Unterbefehl ist) erzeugt einen Menüeintrag, die Teile davor erzeugen den Eintrag des Hauptmenüs bzw. die Untermenüs die zu dem Menüeintrag führen. Einträge im Hauptmenü und Untermenüs werden automatisch bei Bedarf erzeugt, Sie müssen diese nicht manuell anlegen.
<i>Tastenkürzel:</i>	Eine Bitmaske die aus einer Kombination vordefinierter <i>Tasten</i> und <i>Modifikatoren</i> besteht, zusammengefügt mit einem „+“-Zeichen. Die <i>Tasten</i> können entweder eine Ziffer, ein Großbuchstabe, eine Funktionstaste („F1“ ... „F12“) oder ein anderer Tastenname sein der von der Qt-Klasse „QKeySequence“ verstanden wird, inklusive Tastenkürzel für vordefinierte Aktionen (wie zum Beispiel „::Copy“). Typische <i>Modifizierer</i> sind „SHIFT“, „ALT“ und „CTRL“.
<i>ID:</i>	Eine eindeutige ID kann intern verwendet werden um dieses Menü oder diesen Menüeintrag zu identifizieren. Nur Großbuchstaben, Ziffern und „_“ sollten verwendet werden und sie sollte mit „ID_“ beginnen. Es liegt in Ihrer Verantwortung dafür zu sorgen dass die gleiche ID nicht nochmals verwendet wird.

4.13.1.3 Unterbefehle

#checkable:	Macht den Menüeintrag <i>auswählbar</i> , so dass er ein- und ausgeschaltet werden kann.
#disabled:	<i>Deaktiviert</i> ein Menü/Untermenü oder einen Menüeintrag.
#enabled:	<i>Aktiviert</i> ein Menü/Untermenü oder einen Menüeintrag.
#exclusive(Gruppe):	Fügt den Menüeintrag zu einer <i>exklusivenGruppe</i> hinzu (ein Eintrag aus vielen). Die übergebene Gruppe sollte für keinen anderen Zweck verwendet werden. Nur ein Eintrag in dieser Gruppe kann zu einem Zeitpunkt ausgewählt sein.

Das Kwave Handbuch

#group(<i>Liste</i>):	<p>Fügt den Menüeintrag oder Menü/Untermenü zu einer oder mehreren Gruppen hinzu, so dass die Applikation einige Menüeinträge auf einmal aktivieren/deaktivieren kann ohne die IDs der einzelnen Einträge wissen zu müssen. Mehrere Gruppen können als eine per „,“ getrennte Liste übergeben werden. Gruppen-Namen müssen mit einem „@“ beginnen. Die folgenden Gruppen sind vordefiniert:</p> <p>@CLIPBOARD:</p> <p style="margin-left: 20px;">Nur aktiviert wenn die Zwischenablage nicht leer ist.</p>
	<p>@LABELS:</p> <p style="margin-left: 20px;">Nur aktiviert wenn das aktuelle Signal mindestens eine Markierung enthält.</p>
	<p>@NOT_CLOSED:</p> <p style="margin-left: 20px;">Aktiviert wenn das aktuelle Signal nicht geschlossen ist (es darf jedoch leer sein oder die Länge Null haben).</p>
	<p>@SELECTION:</p> <p style="margin-left: 20px;">Aktiviert wenn die Auswahl nicht leer ist (mehr als ein Sample ist ausgewählt).</p>
	<p>@SIGNAL:</p> <p style="margin-left: 20px;">Aktiviert wenn ein Signal vorhanden ist und nicht die Länge Null hat.</p>
#hidden:	<p>Blendet ein Menü/Untermenü oder einen Menüeintrag aus.</p>
#icon(<i>Name</i>):	<p>Weist einem Menüeintrag ein <i>Symbol</i> zu. Der <i>Name</i> des Symbols sollte einer Datei (ohne Pfad und ohne Dateierweiterung) entsprechen die auf Ihrem System oder mit Kwave installiert wurde.</p>

#listmenu(<i>ID,Befehl</i>):	Fügt einen Platzhalter für eine Liste von Menüeinträgen in einem Untermenü hinzu. Die eindeutige <i>ID</i> die in diesem Unterbefehl angegeben wird, wird verwendet um Menüeinträge zur Liste hinzuzufügen, zu entfernen oder die Liste komplett zu leeren. Der Parameter mit dem <i>Befehl</i> muss ein „%1“ als Parameter enthalten, dieser wird durch den Text des jeweiligen Menüeintrags ersetzt wenn der Eintrag aktiviert wurde. Dieser Unterbefehl wird intern verwendet für die Liste der zuletzt geöffneten Dateien, Liste der Spuren und die Fensterliste.
#separator:	Fügt eine Trennlinie in einem Untermenü hinzu.

4.13.2 msgbox

4.13.2.1 Syntax: msgbox(*Text*)

Zeigt ein Nachrichten-Fenster mit einem *Text* und den beiden Schaltflächen **OK** (kehrt ohne Fehlercode zurück) und **Abbrechen** (liefert einen Fehlercode). Sie können diesen Befehl verwenden um dem Anwender eine Möglichkeit zu bieten um ein laufendes Skript abzubrechen.

4.13.2.2 Parameter

<i>Text</i> :	Eine Nachricht die im Nachrichten-Fenster angezeigt wird, sollte eine Frage enthalten die mit OK oder Abbrechen beantwortet werden kann.
---------------	--

4.14 n

4.14.1 newsignal

4.14.1.1 Syntax: newsignal(*Samples, Abtastrate, Bits, Spuren*)

Erzeugt ein neues Signal, mit der angegebenen Länge in *Samples*, einer *Abtastrate* in *Samples* pro Sekunde (Fließkommazahl), einer Anzahl von *Bits* pro Sample und Anzahl von *Spuren*. Sie können die Länge in *Samples* berechnen indem Sie die gewünschte Länge in Sekunden mit der *Abtastrate* multiplizieren.

4.14.1.2 Parameter

<i>Samples</i> :	Länge des Signal in Samples.
<i>Abtastrate</i> :	Abtastrate in Samples pro Sekunde.

<i>Bits:</i>	Anzahl der Bits pro Sample, darf nicht Null sein, sollte eine Zahl zwischen 8 und 32 sein.
<i>Spuren:</i>	Anzahl der Spuren.

4.14.2 next

4.14.2.1 Syntax: `next()`

Entspricht der Werkzeugleisten-Schaltfläche **Nächstes**. Wenn die Wiedergabe aktuell läuft, wird bis zur nächsten Markierung vor gespult. Wenn die Wiedergabe nicht läuft, entspricht dieser Befehl dem Befehl `view:scroll_next_label()`.

4.14.2.2 Siehe auch

[view:scroll_next_label\(\)](#), [prev\(\)](#)

4.15 o

4.15.1 open

4.15.1.1 Syntax: `open(Dateiname)`

Öffnet eine Kwave-Skript- oder Audio-Datei. Wenn kein Dateiname übergeben wurde, so wird ein Dateiauswahl-Dialog geöffnet. Je nach GUI-Modus wird die Datei in einem neuen Unterpfenster (MDI oder Unterfenster) oder in einem neuen Hauptfenster geöffnet (SDI, falls bereits eine Datei geladen ist).

4.15.1.2 Parameter

<i>Dateiname:</i>	Dateiname inklusive Pfad und Dateierweiterung
-------------------	---

4.15.1.3 Siehe auch

[close\(\)](#)

4.15.2 openrecent

4.15.2.1 Syntax: `openrecent(Dateiname)`

Entspricht praktisch dem Befehl `open()`, wird jedoch intern für die Liste der zuletzt geöffneten Dateien im Menü **Datei+Zuletzt geöffnete Dateien**. Der Parameter *Dateiname* ist in diesem Befehl nicht optional.

4.15.2.2 Parameter

Dateiname:

Eintrag der Liste der zuletzt geöffneten Dateien

4.15.2.3 Siehe auch

[open\(\)](#)

4.16 p

4.16.1 paste

4.16.1.1 Syntax: `paste()`

Ersetzt die aktuelle Auswahl mit dem Inhalt der Zwischenablage. Wenn die Zwischenablage leer ist bewirkt diese Funktion nichts. Die Abtastrate der eingefügten Daten wird bei Bedarf an die Abtastrate des aktuellen Signals angepasst. Nur aktivierte Spuren werden beeinflusst. Bedenken Sie, dass dies eine Zeitverschiebung zwischen aktivierten und deaktivierten Spuren bewirken kann! Wenn sich die Anzahl der Spuren zwischen dem aktuellen Signal und dem Inhalt der Zwischenablage unterscheidet, werden die eingefügten Daten gleichmäßig über die aktiven Spuren gemischt.

4.16.1.2 Siehe auch

[copy\(\)](#)

4.16.2 pause

4.16.2.1 Syntax: `continue()`

Entspricht der Werkzeugleisten-Schaltfläche **Pause** und lässt die Wiedergabe anhalten falls sie gerade läuft oder fortsetzen wenn sie angehalten wurde.

4.16.2.2 Siehe auch

[continue\(\)](#)

4.16.3 playback_start

4.16.3.1 Syntax: `playback_start()`

Entspricht der Werkzeugleisten-Schaltfläche **Start** und lässt die Wiedergabe starten wenn sie angehalten ist.

4.16.4 plugin

4.16.4.1 Syntax: `plugin(Name, [Parameter ...])`

Führt ein Modul aus, mit einer optionalen Liste von Parametern. Falls keine Parameterliste angegeben wurde, wird die Einrichtungs-Funktion des Moduls aufgerufen, unter Verwendung der Parameter des letzten Aufrufs oder der Standard-Parameter (normalerweise wird je nach Modul ein Einrichtungs-Dialog angezeigt). Für eine Beschreibung der diversen Module sei auf das Kapitel [Module](#) verwiesen.

4.16.4.2 Parameter

<i>Name:</i>	der (interne) Name eines Kwave-Moduls
<i>Parameter:</i>	eine Liste der vom Modul unterstützten Parameter (optional)

4.16.4.3 Siehe auch

[plugin:execute\(\)](#), [plugin:setup\(\)](#)

4.16.5 plugin:execute

4.16.5.1 Syntax: `plugin:execute(Name, [Parameter ...])`

Ähnlich wie der Befehl `plugin()`, führt jedoch nicht die Setup-Funktion des Moduls aus wenn keine Parameterliste übergeben wurde.

4.16.5.2 Parameter

<i>Name:</i>	der (interne) Name eines Kwave-Moduls
<i>Parameter:</i>	eine Liste von Parametern die vom Modul unterstützt werden

4.16.6 plugin:setup

4.16.6.1 Syntax: `plugin:setup(Name, [Parameter ...])`

Ruft die *Setup*-Funktion eines Moduls auf, mit einer optionalen Parameterliste. Wenn keine Parameter angegeben wurden werden die Parameter des letzten Aufrufs verwendet. Normalerweise wird je nach Modul ein Einrichtungs-Dialog angezeigt. Für eine Beschreibung der diversen Module sei auf das Kapitel [Module](#) verwiesen.

4.16.6.2 Parameter

<i>Name:</i>	der (interne) Name eines Kwave-Moduls
<i>Parameter:</i>	eine Liste der vom Modul unterstützten Parameter (optional)

4.16.7 prev

4.16.7.1 Syntax: prev()

Entspricht der Werkzeugleisten-Schaltfläche **Vorheriges**. Wenn die Wiedergabe aktuell läuft, wird bis zur vorherigen Markierung zurückgespult. Wenn die Wiedergabe nicht läuft, entspricht dieser Befehl dem Befehl `view:scroll_prev_label()`.

4.16.7.2 Siehe auch

[view:scroll_prev_label\(\)](#), [next\(\)](#)

4.17 q

4.17.1 quit

4.17.1.1 Syntax: quit()

Schließt das aktuelle Hauptfenster, inklusive aller Unterfenster. Im SDI-Modus ist dies identisch mit dem Befehl `close()`.

4.17.1.2 Siehe auch

[close\(\)](#)

4.18 r

4.18.1 redo

4.18.1.1 Syntax: redo()

Entspricht der Werkzeugleisten-Schaltfläche **Wiederherstellen** und stellt eine Operation her die zuvor mit `undo()` rückgängig gemacht wurde.

4.18.1.2 Siehe auch

[undo\(\)](#)

4.18.2 redo_all

4.18.2.1 Syntax: redo_all()

Ähnlich wie der Befehl `redo()`, stellt jedoch so viele Operationen wie möglich wieder her.

4.18.2.2 Siehe auch

[undo\(\)](#)

4.18.3 reenable_dna

4.18.3.1 Syntax: reenable_dna()

Einige Nachrichten-Fenster bieten die Möglichkeit zu verhindern dass sie erneut erscheinen („nicht nochmal fragen“). Mit diesem Befehl kann man sie wieder freischalten.

4.18.4 reset_toolbars

4.18.4.1 Syntax: reset_toolbars()

Setzt alle Werkzeugeisten-Einstellungen wie Position, Icon-Größe und Position des Beschreibungstexts wieder auf Standardeinstellungen zurück.

4.18.5 revert

4.18.5.1 Syntax: revert()

Setzt die aktuell geladene Datei auf den letzten gespeicherten Zustand zurück. Alle noch nicht gespeicherten Änderungen gehen verloren.

4.18.6 rewind

4.18.6.1 Syntax: rewind()

Entspricht der Werkzeugeisten-Schaltfläche **Zurückspulen**. Wenn die Wiedergabe aktuell läuft, wird um 1/10 des sichtbaren Bereichs zurückgespult. Wenn die Wiedergabe nicht läuft, entspricht dieser Befehl dem Befehl [view:scroll_left\(\)](#).

4.18.6.2 Siehe auch

[view:scroll_left\(\)](#), [forward\(\)](#)

4.19 s

4.19.1 save

4.19.1.1 Syntax: save()

Entspricht der Werkzeugeisten-Schaltfläche **Speichern**. Speichert die aktuelle Datei wenn sie Veränderungen enthält. Wenn die Datei noch keinen Namen hat (z. B. wenn die Datei gerade erst erzeugt wurde und noch keinen Dateinamen hat), entspricht dieser Befehl dem Aufruf [vonsaveas\(\)](#).

4.19.1.2 Siehe auch

[saveas\(\)](#)

4.19.2 saveas

4.19.2.1 Syntax: `saveas([Dateiname])`

Speichert die aktuell geöffnete Datei unter dem angegebenen Dateinamen. Wenn kein Dateiname als Parameter angegeben wurde, wird ein Dialog angezeigt um einen Ordner auszuwählen und einen Dateinamen einzugeben.

4.19.2.2 Parameter

<i>Dateiname:</i>	Dateiname zum speichern (optional)
-------------------	------------------------------------

4.19.3 saveselect

4.19.3.1 Syntax: `saveselect()`

Dieser Befehl entspricht dem Befehl `save()`, speichert aber nur den ausgewählten Bereich und nur die aktiven Spuren anstelle der gesamten Datei.

4.19.3.2 Siehe auch

[save\(\)](#)

4.19.4 select_gui_type

4.19.4.1 Syntax: `select_gui_type(Modus)`

Wählt einen neuen GUI-Modus, zur Verfügung stehen SDI-, MDI- und Unterfenster-Modus. Bitte beachten Sie dass diese Änderung sofort wirksam wird!

4.19.4.2 Parameter

<i>Dateiname:</i>	Name des Modus muss entweder „SDI“, „MDI“ oder „TAB“ sein.
-------------------	--

4.19.5 select_track:all

4.19.5.1 Syntax: `select_track:all()`

Markiert alle Spuren als „aktiviert“. Dies entspricht einem Aufruf von `select_track:on()` für alle existierenden Spuren.

4.19.5.2 Siehe auch

[select_track:on\(\)](#)

4.19.6 select_track:invert

4.19.6.1 Syntax: select_track:all()

Invertiert den „aktiviert“-Zustand aller Spuren. Dies entspricht dem Aufrufen des Befehls `select_track:toggle()` für alle existierenden Spuren.

4.19.6.2 Siehe auch

[select_track:toggle\(\)](#)

4.19.7 select_track:none

4.19.7.1 Syntax: select_track:none()

Markiert alle Spuren als „deaktiviert“. Dies entspricht einem Aufruf von `select_track:off()` für alle existierenden Spuren.

4.19.7.2 Siehe auch

[select_track:off\(\)](#)

4.19.8 select_track:off

4.19.8.1 Syntax: select_track:off(*index*)

Deaktiviert eine einzelne Spur, so dass sie nicht mehr von allen Operationen beeinflusst wird.

4.19.8.2 Parameter

<i>Index:</i>	Index der Spur, beginnend bei 0
---------------	---------------------------------

4.19.9 select_track:on

4.19.9.1 Syntax: select_track:on(*index*)

Aktiviert eine einzelne Spur, so dass sich alle Operationen darauf auswirken.

4.19.9.2 Parameter

Index: | Index der Spur, beginnend bei 0

4.19.10 select_track:toggle

4.19.10.1 Syntax: select_track:toggle(*Index*)

Aktiviert eine deaktivierte Spur oder deaktiviert eine aktivierte Spur.

4.19.10.2 Parameter

Index: | Index der Spur, beginnend bei 0

4.19.11 selectall

4.19.11.1 Syntax: selectall()

Setzt den ausgewählten Bereich auf das komplette Signal, vom ersten bis zum letzten Sample.

4.19.12 selectnext

4.19.12.1 Syntax: selectnext()

Setzt die Auswahl auf den nächsten Bereich nach der aktuellen Auswahl, mit gleicher Länge wie die aktuelle Auswahl. Die Auswahl wird automatisch auf das Ende des Signals beschränkt. Beispiel: wenn Sie den Bereich von 1000 ... 1019 ausgewählt hatten, dann wird die resultierende Auswahl der Bereich von 1020 ... 1039 sein.

4.19.12.2 Siehe auch

[selectprev\(\)](#)

4.19.13 selectnextlabels

4.19.13.1 Syntax: selectnextlabels()

Wählt einen Bereich zwischen den nächsten beiden Markierung hinter der aktuellen Auswahl aus. Wenn nichts ausgewählt ist, wird der Bereich vom Anfang des Signals bis zur ersten Markierung ausgewählt. Ansonsten wird der linke Rand der neuen Auswahl auf die Markierung nach der aktuellen Auswahl (oder auf die letzte Markierung, falls nach der aktuellen Auswahl keine weiteren Markierungen liegen) und der rechte Rand der neuen Auswahl wird auf die Markierung nach dem linken Rand der aktuellen Auswahl gesetzt (oder auf das Ende des Signals wenn dort keine Markierung existiert). Dieser Befehl gibt einen Fehler zurück wenn keine einzigen Markierung vorhanden ist.

4.19.13.2 Siehe auch

[selectprevlabels\(\)](#)

4.19.14 selectnone

4.19.14.1 Syntax: selectnone()

Setzt die Auswahl auf Länge Null.

4.19.15 selectprev

4.19.15.1 Syntax: selectprev()

Setzt die Auswahl auf den nächsten Bereich vor der aktuellen Auswahl, mit gleicher Länge wie die aktuelle Auswahl. Die Auswahl wird automatisch auf den Anfang des Signals beschränkt. Beispiel: wenn Sie den Bereich von 1000 ... 1019 ausgewählt hatten, dann wird die resultierende Auswahl der Bereich von 980 ... 999 sein.

4.19.15.2 Siehe auch

[selectnext\(\)](#)

4.19.16 selectprevlabels

4.19.16.1 Syntax: selectprevlabels()

Wählt einen Bereich zwischen den beiden vorherigen Markierung vor der aktuellen Auswahl aus. Wenn nichts ausgewählt ist, wird der Bereich vom Anfang des Signals bis zur ersten Markierung ausgewählt. Ansonsten wird der rechte Rand der neuen Auswahl auf die Markierung vor der aktuellen Auswahl (oder der ersten Markierung, falls vor der aktuellen Auswahl keine weiteren Markierungen liegen) und der linke Rand der neuen Auswahl wird auf die Markierung vor dem rechten Rand der aktuellen Auswahl gesetzt (oder auf den Start des Signals, wenn dort keine Markierung existiert). Dieser Befehl gibt einen Fehler zurück wenn keine einzige Markierung vorhanden ist.

4.19.16.2 Siehe auch

[selectnextlabels\(\)](#)

4.19.17 selecttopleft

4.19.17.1 Syntax: selecttopleft()

Setzt den Start der Auswahl auf den Start des Signals, das Ende der Auswahl bleibt unverändert.

4.19.17.2 Siehe auch

[selecttright\(\)](#)

4.19.18 selecttoright

4.19.18.1 Syntax: selecttoright()

Setzt das Ende der Auswahl auf das Ende des Signals, der Anfang der aktuellen Auswahl bleibt unverändert.

4.19.18.2 Siehe auch

[selecttoleft\(\)](#)

4.19.19 selectvisible

4.19.19.1 Syntax: selectvisible()

Setzt die Auswahl auf den im aktuellen Fenster angezeigten Bereich.

4.19.20 start

4.19.20.1 Syntax: start()

Entspricht der Werkzeugleisten-Schaltfläche **Start** und lässt die Wiedergabe von Anfang an starten wenn sie angehalten ist oder weiterlaufen falls sie aktuell pausiert wurde.

4.19.20.2 Siehe auch

[stop\(\)](#)

4.19.21 stop

4.19.21.1 Syntax: stop()

Entspricht der Werkzeugleisten-Schaltfläche **Stop** und lässt die Wiedergabe anhalten wenn sie gerade läuft.

4.19.21.2 Siehe auch

[start\(\)](#)

4.19.22 sync

4.19.22.1 Syntax: sync()

Wartet bis sich alle Befehle die asynchron gestartet wurden beendet haben. Falls die Warteschlange für verzögerte Ausführung aktuell leer ist hat dieser Befehl keine Auswirkung.

4.19.22.2 Siehe auch

[delayed\(\)](#)

4.20 u

4.20.1 undo

4.20.1.1 Syntax: undo()

Entspricht der Werkzeugleisten-Schaltfläche **Rückgängig** und macht die letzte Operation rückgängig.

4.20.1.2 Siehe auch

[redo\(\)](#)

4.20.2 undo_all

4.20.2.1 Syntax: undo_all()

Ähnlich wie der Befehl **undo()**, stellt jedoch so viele Operationen wie möglich wieder her.

4.20.2.2 Siehe auch

[undo\(\)](#)

4.21 v

4.21.1 view:scroll_end

4.21.1.1 Syntax: view:scroll_end()

Verschiebt die aktuelle Ansicht an das *Ende* des Signals.

4.21.1.2 Siehe auch

[view:scroll_start\(\)](#)

4.21.2 view:scroll_left

4.21.2.1 Syntax: view:scroll_left()

Verschiebt die aktuelle Ansicht um 1/10 der Breite des aktuell sichtbaren Bereichs in Richtung Anfang des Signals. Wenn der Anfang des Signals erreicht wurde, beginnt die Ansicht mit dem Anfang des Signals.

4.21.2.2 Siehe auch

[view:scroll_right\(\)](#)

4.21.3 **view:scroll_next**

4.21.3.1 Syntax: **view:scroll_next()**

Verschiebt die aktuelle Ansicht in Richtung des Endes des Signals, um die Breite des aktuell sichtbaren Bereichs.

4.21.3.2 Siehe auch

[view:scroll_prev\(\)](#)

4.21.4 **view:scroll_next_label**

4.21.4.1 Syntax: **view:scroll_next_label()**

Verschiebt die Ansicht nach rechts und versucht auf die Position der nächsten Markierung zu zentrieren. Wenn rechts von der aktuellen Position keine Markierung liegt, wird die Ansicht an das Ende des Signals verschoben.

4.21.4.2 Siehe auch

[view:scroll_prev_label\(\)](#)

4.21.5 **view:scroll_prev**

4.21.5.1 Syntax: **view:scroll_prev()**

Verschiebt die aktuelle Ansicht in Richtung des Anfangs des Signals, um die Breite des aktuell sichtbaren Bereichs.

4.21.5.2 Siehe auch

[view:scroll_next\(\)](#)

4.21.6 **view:scroll_prev_label**

4.21.6.1 Syntax: **view:scroll_prev_label()**

Verschiebt die Ansicht nach links und versucht auf die Position der vorhergehenden Markierung zu zentrieren. Wenn links von der aktuellen Position keine Markierung liegt, wird die Ansicht auf den Start des Signals verschoben.

4.21.6.2 Siehe auch

[view:scroll_next_label\(\)](#)

4.21.7 view:scroll_right

4.21.7.1 Syntax: view:scroll_right()

Verschiebt die aktuelle Ansicht um 1/10 der Breite des aktuell sichtbaren Bereichs in Richtung Ende des Signals. Wenn das Ende des Signals erreicht wurde, endet die Ansicht mit dem Ende des Signals.

4.21.7.2 Siehe auch

[view:scroll_left\(\)](#)

4.21.8 view:scroll_start

4.21.8.1 Syntax: view:scroll_start()

Verschiebt die aktuelle Ansicht an den Anfang des Signals.

4.21.8.2 Siehe auch

[view:scroll_end\(\)](#)

4.21.9 view:zoom_all

4.21.9.1 Syntax: view:zoom_all()

Passt den Vergrößerungsfaktor so an dass das Signal vollständig in der aktuellen Ansicht angezeigt wird.

4.21.10 view:zoom_in

4.21.10.1 Syntax: view:zoom_in([*Position*])

Reduziert den Vergrößerungsfaktor (in Samples pro Pixel) um 30%, so dass mehr Details sichtbar werden. Wenn eine *Position* angegeben wurde, wird versucht die Ansicht auf diese Position zu zentrieren, ansonsten wird die Mitte der Ansicht vor der Zoom-Änderung zum zentrieren verwendet. Der Vergrößerungsfaktor wird auf ein Minimum von fünf Samples pro Breite der Ansicht begrenzt.

4.21.10.2 Parameter

Position:

eine mit Null beginnende Position in Samples auf die die Ansicht zentriert werden soll (optional)

4.21.10.3 Siehe auch

[view:zoom_out\(\)](#)

4.21.11 view:zoom_normal

4.21.11.1 Syntax: `view:zoom_normal()`

Setzt den Vergrößerungsfaktor auf ein Pixel pro Sample (Faktor 1,0) und versucht die aktuelle Zentrierung der Ansicht beizubehalten.

4.21.12 view:zoom_out

4.21.12.1 Syntax: `view:zoom_out([Position])`

Erhöht den Vergrößerungsfaktor (in Samples pro Pixel) um 30%, so dass weniger Details sichtbar sind. Wenn eine *Position* angegeben wurde, wird versucht die Ansicht auf diese Position zu zentrieren, ansonsten wird die Mitte der Ansicht vor der Zoom-Änderung zum zentrieren verwendet. Der maximale Vergrößerungsfaktor wird durch die Anzahl der Samples des gesamten Signals und der Breite der Ansicht begrenzt.

4.21.12.2 Parameter

Position:

eine mit Null beginnende Position in Samples auf die die Ansicht zentriert werden soll (optional)

4.21.12.3 Siehe auch

[view:zoom_in\(\)](#)

4.21.13 view:zoom_selection

4.21.13.1 Syntax: `view:zoom_selection()`

Passt die Ansicht Vergrößerungsfaktor und Start des sichtbaren Bereichs) so an dass sie der aktuellen Auswahl entspricht. Dieser Befehl bewirkt nichts wenn die Auswahl leer ist.

4.22 w

4.22.1 window:activate

4.22.1.1 Syntax: `window:activate([Überschrift])`

Aktiviert ein Unterfenster, das über seine *Überschrift* identifiziert wird. Wenn das Unterfenster minimiert ist wird es wiederhergestellt. Nur verfügbar wenn im MDI- oder Unterfenster-Modus. Dieser Befehl wird intern vom **Fenster**-Menü verwendet.

4.22.1.2 Parameter

<i>Überschrift:</i>	die Überschrift des Unterfensters das aktiviert werden soll
---------------------	---

4.22.2 window:cascade

4.22.2.1 Syntax: `window:cascade()`

Kaskadiert alle Unterfenster wenn im MDI-Modus. Alle Unterfenster die aktuell minimiert sind bleiben minimiert, sie werden nicht wiederhergestellt.

4.22.3 window:click

4.22.3.1 Syntax: `window:click(Klasse, x, y)`

Sendet einen Mausklick-Ereignis an ein Fenster das über seine *Klasse* identifiziert wird. Das Ereignis wird nur zu dem ersten Fenster gesendet das den angegebenen Klassen-Namen hat, deshalb sollten Sie sicher stellen dass nur eine einzige Instanz des angegebenen Fensters existiert wenn der Befehl ausgeführt wird.

4.22.3.2 Parameter

<i>Klasse:</i>	Name der Fensterklasse
<i>x:</i>	X-Position, relativ zum linken Rand des Fensters (in Pixel)
<i>y:</i>	Y-Position, relativ zur Oberkante des Fensters (in Pixel)

4.22.4 window:close

4.22.4.1 Syntax: `window:close(Klasse)`

Schließt ein Fenster, das über seine *Klasse* identifiziert wird. Nur das erste Fenster mit dem angegebenen Klassen-Namen wird geschlossen, deshalb sollten Sie sicher stellen dass nur eine einzige Instanz des angegebenen Fensters existiert wenn der Befehl ausgeführt wird.

4.22.4.2 Parameter

<i>Klasse:</i>	Name der Fensterklasse
----------------	------------------------

4.22.5 window:minimize

4.22.5.1 Syntax: window:minimize

Minimiert das aktuell aktive Unterfenster im MDI-Modus oder das aktuelle Hauptfenster im SDI- oder Unterfenster-Modus.

4.22.6 window:mousemove

4.22.6.1 Syntax: window:resize(*Klasse*, *x*, *y*)

Sendet einen Mausbewegungs-Ereignis an ein Fenster das über seine *Klasse* identifiziert wird. Das Ereignis wird nur zu dem ersten Fenster gesendet das den angegebenen Klassen-Namen hat, deshalb sollten Sie sicher stellen dass nur eine einzige Instanz des angegebenen Fensters existiert wenn der Befehl ausgeführt wird.

4.22.6.2 Parameter

<i>Klasse</i> :	Name der Fensterklasse
<i>x</i> :	X-Position, relativ zum linken Rand des Fensters (in Pixel)
<i>y</i> :	Y-Position, relativ zur Oberkante des Fensters (in Pixel)

4.22.7 window:next_sub

4.22.7.1 Syntax: window:next_sub()

Aktiviert das *nächste* Unterfenster wenn im MDI- oder Unterfenster-Modus. Wenn das nächste Unterfenster minimiert war, wird es wiederhergestellt.

4.22.8 window:prev_sub

4.22.8.1 Syntax: window:prev_sub()

Aktiviert das *vorhergehende* Unterfenster wenn im MDI- oder Unterfenster-Modus. Wenn das vorhergehende Unterfenster minimiert war, wird es wiederhergestellt.

4.22.9 window:resize

4.22.9.1 Syntax: window:resize(*Klasse*, *Breite*, *Höhe*)

Ändert die Größe eines Fensters, das anhand seiner *Klasse* identifiziert wird auf eine neue *Breite* und *Höhe*. Die Änderung wirkt sich nur auf das erste Fenster mit dem angegebenen Klassen-Namen aus, deshalb sollten Sie sicher stellen dass nur eine einzige Instanz des angegebenen Fensters existiert wenn der Befehl ausgeführt wird.

4.22.9.2 Parameter

<i>Klasse:</i>	Name der Fensterklasse
<i>Breite:</i>	neue Breite des Fensters (in Pixel)
<i>Höhe:</i>	neue Höhe des Fensters (in Pixel)

4.22.10 window:screenshot

4.22.10.1 Syntax: `window:screenshot(Klasse, Dateiname)`

Erstellt ein Bildschirmfoto eines Fensters, das über seine *Klasse* identifiziert wird und speichert es in einer Datei. Das Bildschirmfoto wird nur vom ersten Fenster erstellt das den angegebenen Klassen-Namen hat, deshalb sollten Sie sicher stellen dass nur eine einzige Instanz des angegebenen Fensters existiert wenn der Befehl ausgeführt wird. Das Dateiformat ist aktuell festgelegt und muss *.png sein.

4.22.10.2 Parameter

<i>Klasse:</i>	Name der Fensterklasse
<i>Dateiname:</i>	Dateiname unter dem das Bildschirmfoto gespeichert werden soll, muss die Erweiterung *.png haben

4.22.11 window:sendkey

4.22.11.1 Syntax: `window:sendkey(Klasse, Tastenkürzel)`

Sendet ein Tastendruck- und Loslassen-Ereignis an ein Fenster das über seine *Klasse* identifiziert wird. Die Ereignisse werden nur zu dem ersten Fenster gesendet das den angegebenen Klassen-Namen hat, deshalb sollten Sie sicher stellen dass nur eine einzige Instanz des angegebenen Fensters existiert wenn der Befehl ausgeführt wird.

4.22.11.2 Parameter

<i>Klasse:</i>	Name der Fensterklasse
<i>Tastenkürzel:</i>	Das zu sendende Tastenkürzel, gleiche Syntax wie beim Einrichten der Menüs

4.22.11.3 Siehe auch

Beschreibung des Parameters *Tastenkürzel* des Befehls `menu()`.

4.22.12 **window:title**

4.22.12.1 Syntax: **window:title()**

Kachelt alle Unterfenster wenn im MDI-Modus, auf eine von Plasma vorgegebene Art und Weise. Alle aktuell minimierten Unterfenster bleiben minimiert, sie werden nicht wiederhergestellt.

4.22.13 **window:title_vertical**

4.22.13.1 Syntax: **window:title_vertical()**

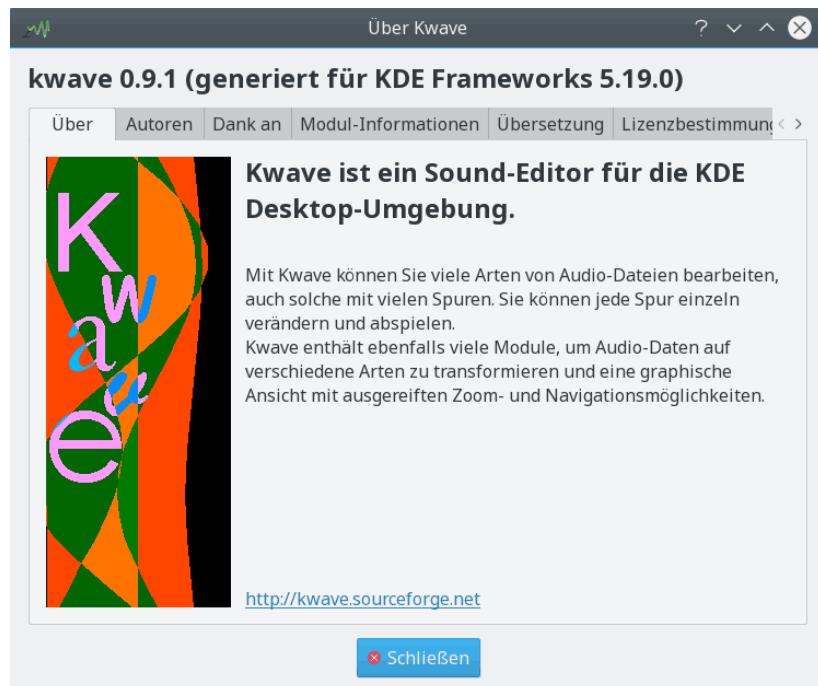
Ordnet alle Unterfenster untereinander an wenn im MDI-Modus. Alle aktuell minimierten Unterfenster bleiben minimiert, sie werden nicht wiederhergestellt.

Kapitel 5

Module

5.1 Modul-Referenz

5.2 about (Über Kwave)



Interner Name:

about

Modul-Typ:

GUI

Beschreibung:

Zeigt ein Dialogfenster mit mehreren Karteireitern, mit folgenden Informationen:

- Allgemeine Informationen über das Programm
- Autoren, Beitragende und Copyright-Halter
- alle gefundenen Module einschließlich ihrer Versionsnummern und Autoren
- Informationen über das Übersetzer-Team
- Copyright- und Lizenz-Informationen

5.3 amplifyfree (Frei Verstärken)



Interner Name:

Frei Verstärken

Modul-Typ:

Effekt

Beschreibung:

Verstärkt die aktuelle Auswahl mit einer Kurve die aus einem Satz von Koordinaten und einer Interpolations-Methode besteht. Sowohl die Koordinaten auf der Zeitachse als auch auf der Amplituden-Achse müssen auf den Bereich zwischen 0,0 und 1,0 normiert sein.

Parameter

Operation

Interner Name, für die Behandlung von Rückgängig/Wiederherstellen. Mögliche Werte sind:

Schlüsselwort	Beschreibung
fade in	Einblenden, Kurve von 0,0/0,0 nach 1,0/1,0

Das Kwave Handbuch

fade out	Ausblenden, Kurve von 0,0/1,0 nach 1,0/0,0
fade intro	Intro-Einblenden, eine Sekunde Pause, dann Einblenden
fade outro	Outro-Ausblenden, erst Ausblenden, dann eine Sekunde Pause
amplify free	benutzerdefinierte Kurve

Interpolation

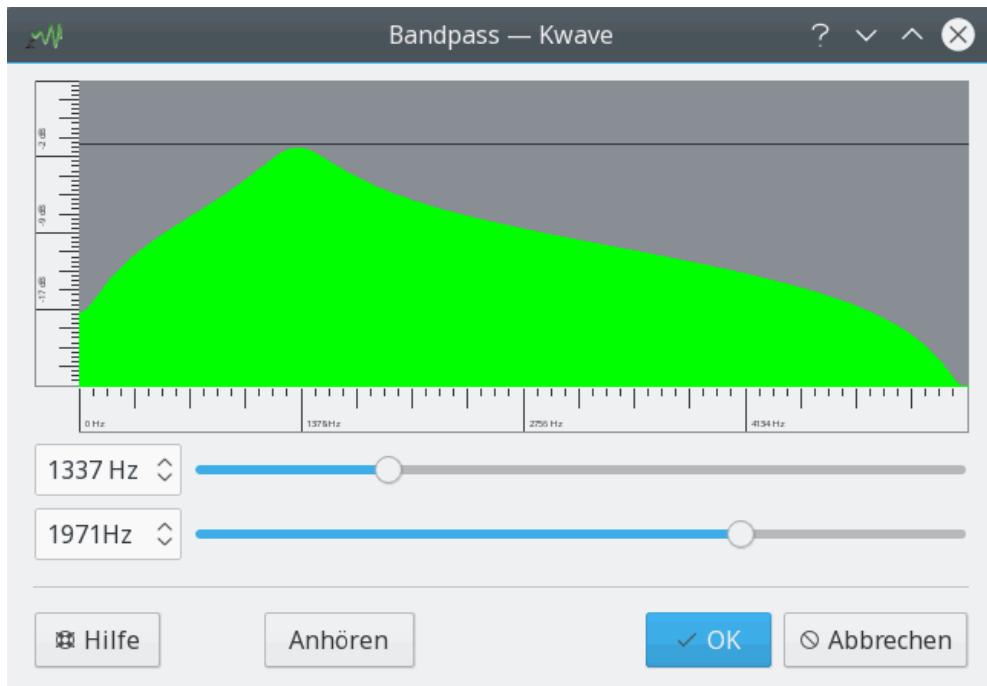
Interpolations-Art, mögliche Werte sind:

Schlüsselwort	Beschreibung
linear	Linear
spline	Spline
n-polynom	Polynom, n-ter Grad
3-polynom	Polynom, dritter Grad
5-polynom	Polynom, 5ter Grad
7-polynom	Polynom, 7ter Grad
sample_hold	Sample + Hold

Kurve

Eine per Komma getrennte Liste von Koordinaten, normiert zwischen 0,0 und 1,0, müssen (aufsteigend) nach Zeitachse sortiert sein, beginnend mit der Zeit 0,0 und endend mit der Zeit 1,0.

5.4 band_pass (Bandpassfilter)



Interner Name:

band_pass

Modul-Typ:

Effekt

Beschreibung:

Wendet einen einfachen Bandpassfilter auf die aktuelle Auswahl an. Ein Bandpass lässt einen Frequenzbereich um eine *Mittenfrequenz* herum passieren und filtert alle Frequenzen aus die mehr als die Hälfte der *Bandbreite* über- oder unterhalb der Mittenfrequenz liegen.

Der Filter ist zweiten Grades und ist implementiert wie im Buch "*An introduction to digital filter theory*" von Julius O. Smith und in Moore's Buch beschrieben, wobei die normalisierte Version von Moore's Buch verwendet wird.

Parameter:

Frequenz

Mittenfrequenz des Filters in Hz, muss unterhalb der halben Abtastrate der Datei liegen.

Bandbreite

Bandbreite des Filters in Hz.

5.5 codec_ascii (ASCII Codec)

Interner Name:

codec_ascii

Modul-Typ:

Codec

Unterstützte Dateitypen:

Beschreibung:	ASCII encoded audio
Datei-Erweiterungen:	*.ascii
MIME-Typen:	audio/x-audio-ascii

Unterstützte Metadataten:

(alle bekannten Datei-Informationen, siehe Abschnitt)

5.6 codec_audiofile (Audiofile Codec)

Interner Name:

codec_audiofile

Modul-Typ:

Codec [nur Import]

Unterstützte Dateitypen:

Das Kwave Handbuch

Beschreibung:	Amiga IFF/8SVX Sound File Format
Datei-Erweiterungen:	*.8svx
MIME-Typen:	audio/x-8svx

Beschreibung:	NeXT, Sun Audio
Datei-Erweiterungen:	*.au, *.snd
MIME-Typen:	audio/basic

Beschreibung:	Compressed Audio Interchange Format
Datei-Erweiterungen:	*.aifc
MIME-Typen:	audio/x-aifc

Beschreibung:	Audio Interchange Format
Datei-Erweiterungen:	*.aif, *.aiff
MIME-Typen:	audio/x-aiff

Beschreibung:	Audio Visual Research File Format
Datei-Erweiterungen:	*.avr
MIME-Typen:	audio/x-avr

Beschreibung:	Core Audio File Format
Datei-Erweiterungen:	*.caf
MIME-Typen:	audio/x-caf

Beschreibung:	Berkeley, IRCAM, Carl Sound Format
Datei-Erweiterungen:	*.sf
MIME-Typen:	audio/x-ircam

Beschreibung:	NIST SPHERE Audio File Format
Datei-Erweiterungen:	*.nist
MIME-Typen:	audio/x-nist

Beschreibung:	Sample Vision Format
Datei-Erweiterungen:	*.smp
MIME-Typen:	audio/x-smp

Beschreibung:	Creative Voice
Datei-Erweiterungen:	*.voc
MIME-Typen:	audio/x-voc

Unterstützte Metadataten:
(keine)

5.7 codec_flac (FLAC Codec)

Interner Name:
codec_flac

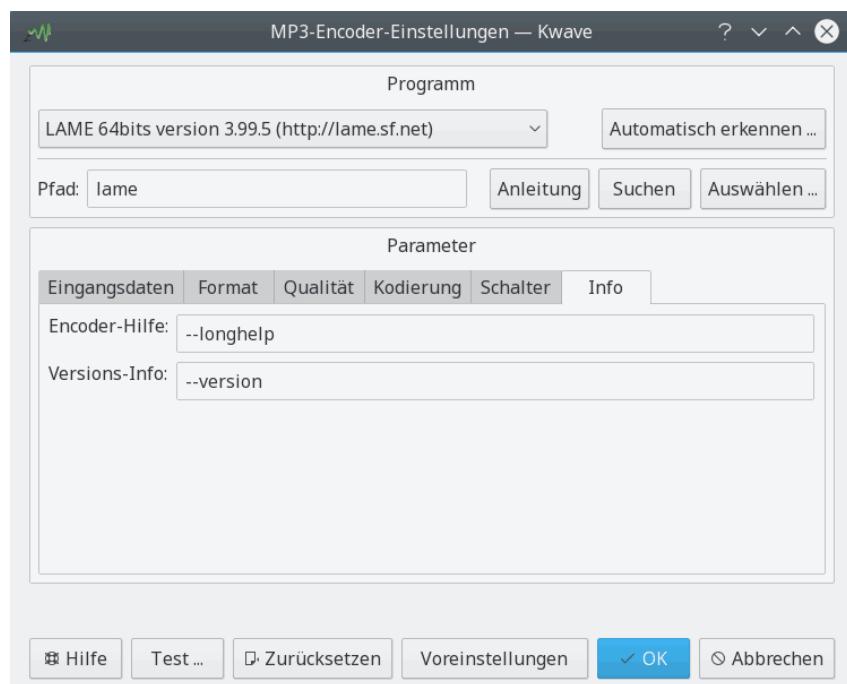
Modul-Typ:
Codec

Unterstützte Dateitypen:

Beschreibung:	FLAC Audio (Free Lossless Audio Codec)
Datei-Erweiterungen:	*.flac
MIME-Typen:	audio/x-flac

Unterstützte Metadataten:
[Datum](#), [Name](#), [Version](#), [Album](#), [Stück](#), [Autor](#), [Darsteller](#), [Copyright](#), [Lizenz](#), [Organisation](#),
[Thema](#), [Genre](#), [Quelle](#), [Kontakt](#), [ISRC](#), [Software](#), [Ton-Techniker](#), [Basis-Qualität](#)

5.8 codec_mp3 (MP3 Codec)



Interner Name:

codec_mp3

Modul-Typ:

Codec

Unterstützte Dateitypen:

Beschreibung:	MPEG Layer III Audio
Datei-Erweiterungen:	*.mp3
MIME-Typen:	audio/x-mp3, audio/mpeg\$

Beschreibung:	MPEG Layer II Audio
Datei-Erweiterungen:	*.mp2
MIME-Typen:	audio/x-mp2, audio/mpeg

Beschreibung:	MPEG Layer I Audio
Datei-Erweiterungen:	*.mp1, *.mpg, *.mpga
MIME-Typen:	audio/x-mpga, audio/mpeg

Unterstützte Metadataten:

Album, Anmerkung, Autor, CD, CDs, Kommentare, Genehmigt, Kontakt, Copyright, Datum, Genre, ISRC, Länge, Lizenz, Medium, Name, Organisation, Darsteller, Software, Techniker, Stück, Stücke, Version

5.9 codec_ogg (Ogg Codec)

Interner Name:

codec_ogg

Modul-Typ:

Codec

Unterstützte Dateitypen:

Beschreibung:	Ogg Opus Audio
Datei-Erweiterungen:	*.opus
MIME-Typen:	audio/ogg, application/ogg, audio/opus

Das Kwave Handbuch

Beschreibung:	Ogg Vorbis Audio
Datei-Erweiterungen:	*.ogg
MIME-Typen:	audio/ogg, audio/x-ogg, application/x-ogg, audio/x-vorbis+ogg

Unterstützte Metadataten:

Album, Autor, Kontakt, Copyright, Datum, Ton-Techniker, Genre, ISRC, Lizenz, Name, Organisation, Darsteller, Software, Quelle, Thema, Stück, Basis-Qualität, Version,

5.10 codec_wav (WAV Codec)

Interner Name:

codec_wav

Modul-Typ:

Codec

Unterstützte Dateitypen:

Beschreibung:	WAV audio
Datei-Erweiterungen:	*.wav
MIME-Typen:	audio/x-wav, audio/vnd.wave, audio/wav

Unterstützte Metadataten:

Album, Anmerkung, Archivierungs-Ort, Autor, CD, Kommentare, Genehmigt, Kontakt, Copyright, Datum, Ton-Techniker, Genre, ISRC, Schlüsselworte, Lizenz, Medium, Name, Organisation, Darsteller, Produkt, Software, Quelle, Ursprungs-Form, Thema, Techniker, Stück, Version,

5.11 debug (Debug-Funktionen)

Interner Name:

debug

Modul-Typ:

Funktion

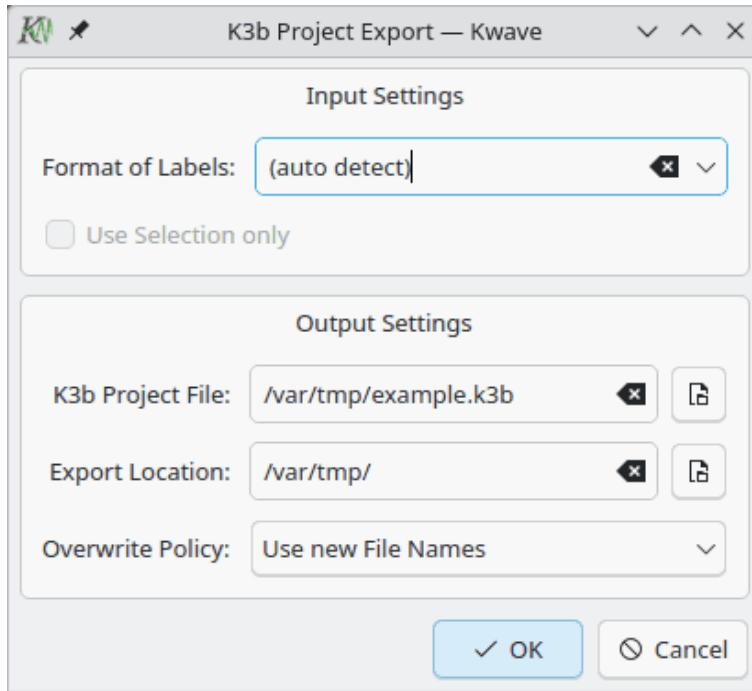
Beschreibung:

Stellt diverse interne Befehle zur Fehlersuche und Automatisierung von Kwave zur Verfügung. Diese sind nur über Menüeinträge erreichbar wenn Kwave im Debug-Modus übersetzt wurde (generiert mit der Option CMAKE_WITH_DEBUG).

Befehle:

.....

5.12 export_k3b (Export als K3b-Projekt)



Interner Name:

export_k3b

Modul-Typ:

Funktion

Beschreibung:

Speichert alle Abschnitte zwischen Markierungen in eine eigene Datei und erzeugt ein K3b-Projekt. Nachdem alle Dateien erfolgreich geschrieben wurden ist es möglich K3b zu starten und das Ergebnis als Audio-CD zu brennen. Dies ist ganz nützlich um eine Aufnahme die aus mehreren Teilen besteht, die durch Markierungen getrennt sind, zu zerteilen und dann als Audio-CD mit mehreren Stücken zu brennen, inklusive CD-Text Metadaten die aus der Beschreibung der Markierungen extrahiert wurden.

(Dieses Modul verwendet intern das -Modul.)

Parameter:

Dateiname

Name der K3b-Projekt-Datei, wird als Basis für die Dateinamen der exportierten Abschnitte verwendet.

Muster

Ein Muster das verwendet wird um den Titel und den Künstlers aus dem Beschreibungstext am Anfang des Abschnitts zu bestimmen. Die folgenden Platzhalter werden beim Erzeugen der CD-Text Metadaten durch ihren entsprechenden Inhalt ersetzt:

Platzhalter	Beschreibung
[%artist]	Wird durch den Namen des Künstlers oder alternativ des Autors des zugehörigen Abschnitts ersetzt.

Das Kwave Handbuch

<p>[%title]</p>	<p>Wird mit dem Titel des Blocks ersetzt, der aus dem Beschreibungstext der Markierung am <i>Anfang</i> des Blocks entnommen wird. Falls dieser Text fehlt, wird auf den Titel der Datei ausgewichen (siehe Datei-Information Name). Sollte dieser ebenfalls fehlen, wird auf den oben beschriebenen Basis-Dateinamen ausgewichen.</p>
------------------------	---

Beispiel: „[%title] ([%artist])“ wird den Künstler „**Beethoven**“ und den Titel „**Symphony No. 5**“ aus der Zeichenfolge „**Symphony No. 5 (Beethoven)**“ erkennen.

nur die Auswahl

Wert	Beschreibung
0	Alle Abschnitte der gesamten Datei speichern.
1	Speichert nur die Abschnitte die in der aktuellen Auswahl liegen. Wenn nichts ausgewählt ist wird die ganze Datei gespeichert.

Speicherort

Bestimmt wo die Abschnitte gespeichert werden sollen.

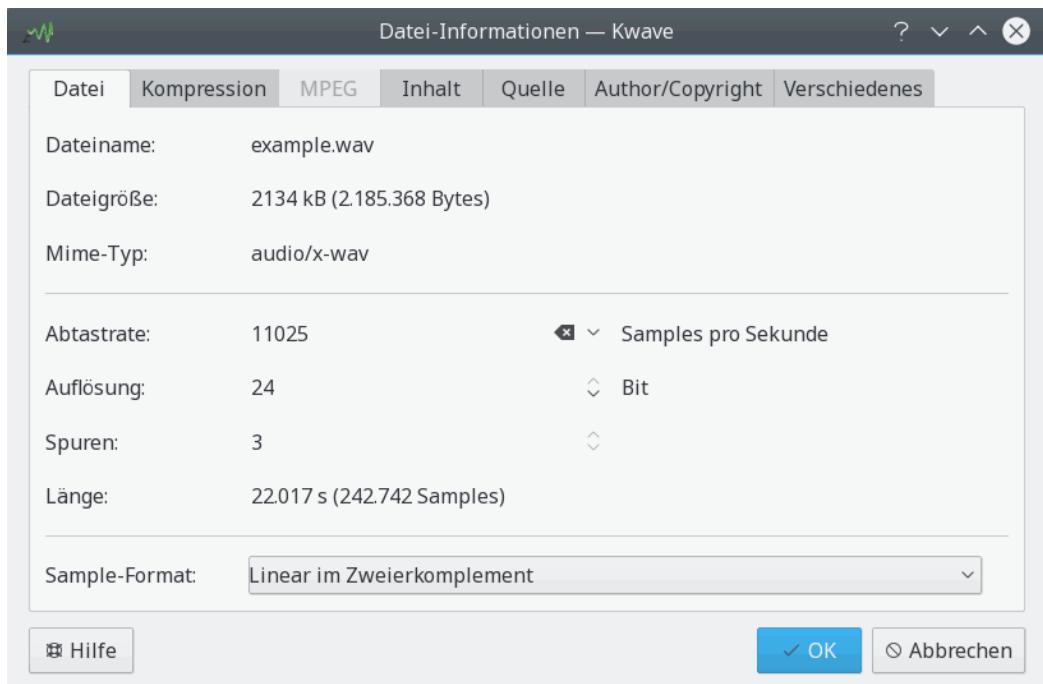
Wert	Beschreibung
0	Speichere in das selbe Verzeichnis wie die K3b-Projektdatei.
1	Speichert in einen Ordner des Verzeichnisses der K3b-Projektdatei, wobei der Dateiname des K3b-Projektdatei als Basis genommen und die Erweiterung „.dir“ angehängt wird.

Überschreib-Regel

Bestimmt wo die Nummerierung beginnen soll.

Wert	Beschreibung
0	Den Index immer mit 1 starten, mit dem Risiko dass existierende Dateien überschrieben werden.
1	Nach dem höchsten gefundenen Index fortsetzen, dies verhindert dass existierende Dateien überschrieben werden.

5.13 fileinfo (Datei-Informationen)



Interner Name:

fileinfo

Modul-Typ:

GUI

Beschreibung:

Zeigt ein Dialogfenster um die Einstellungen und Metadaten der aktuell geöffneten Datei anzuzeigen und zu bearbeiten. Siehe Abschnitt in diesem Handbuch.

5.14 goto (Gehe zu Position)



Interner Name:

goto

Modul-Typ:

Funktion

Beschreibung:

Zeigt ein Dialogfenster mit der Möglichkeit die aktuell ausgewählte Position zu setzen, die als Zeit in Millisekunden, als Position in Samples oder als Prozentsatz der aktuellen Länge der Datei angegeben wird.

Befehle:

Parameter:

Modus

Wert	Beschreibung
0	Position wird in Millisekunden angegeben
1	Position ist in Samples angegeben
2	Position wird als Prozentsatz der Länge der Datei angegeben

Position

Position an die gesprungen werden soll, in Millisekunden, Samples oder Prozentsatz der Länge der Datei, in Abhängigkeit des Parameters *Modus*.

5.15 insert_at (Einfügen an)



Interner Name:

insert_at

Modul-Typ:

Funktion

Beschreibung:

Ähnlich wie das Modul , zeigt jedoch einen Dialog mit der Möglichkeit den aktuellen Inhalt der Zwischenablage an einer angegebenen Stelle einzufügen, die als Zeit in Millisekunden, als Position in Samples oder als Prozentsatz der aktuellen Länge der Datei angegeben wird.

Befehle:

Parameter:

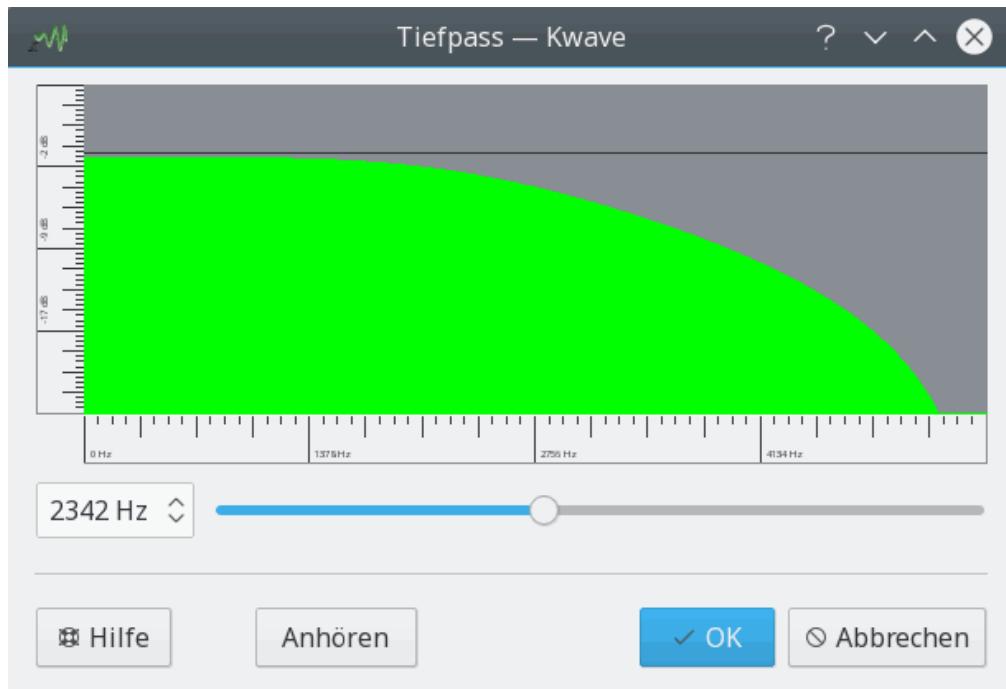
Modus

Wert	Beschreibung
0	Position wird in Millisekunden angegeben
1	Position ist in Samples angegeben
2	Position wird als Prozentsatz der Länge der Datei angegeben

Position

Position an die der Inhalt der Zwischenablage eingefügt werden soll, in Millisekunden, Samples oder Prozentsatz der Länge der Datei, in Abhängigkeit des Parameters *odus*.

5.16 lowpass (Tiefpassfilter)



Interner Name:

lowpass

Modul-Typ:

Effekt

Beschreibung:

Wendet einen einfachen Tiefpassfilter auf die aktuelle Auswahl an. Ein Tiefpassfilter lässt Frequenzen unterhalb einer *Grenzfrequenz* passieren und filtert Frequenzen oberhalb der Grenzfrequenz aus.

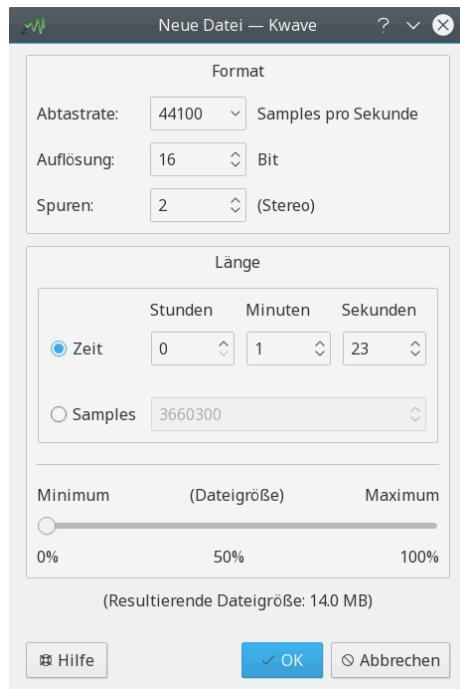
Der Filter ist zweiten Grades und ist implementiert wie im Buch *"The manifold joys of conformal mapping, applications to digital filtering in the studio"* von James A. Moorer (JAES, Vol. 31, No. 11, 1983 November) beschrieben.

Parameter:

Frequenz

Die Grenzfrequenz des Tiefpassfilters in Hz.

5.17 newsignal (Neue Datei)



Interner Name:

newsignal

Modul-Typ:

Funktion

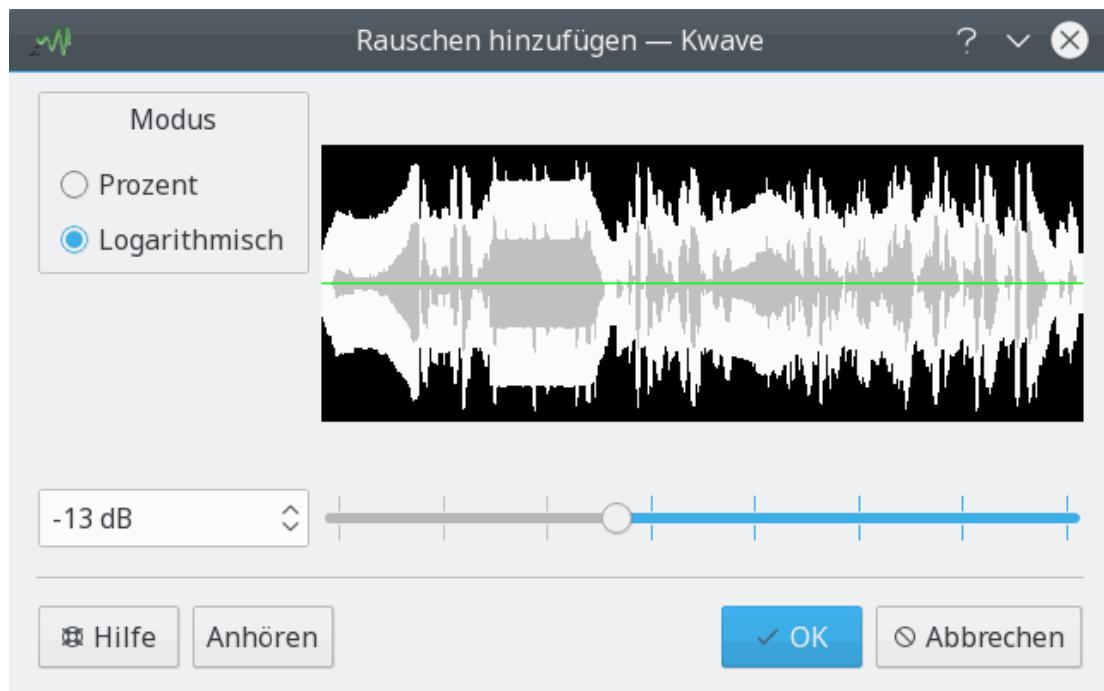
Beschreibung:

Zeigt ein Dialogfenster um eine neue Datei zu erzeugen. Bitte lesen Sie im Abschnitt in diesem Handbuch für weitere Informationen.

Befehle:

,

5.18 noise (Rauschgenerator)


Interner Name:

noise

Modul-Typ:

Effekt

Beschreibung:

Fügt etwas weißes Rauschen zu aktuellen Auswahl hinzu. Die Menge des Rauschens kann zwischen Null (kein Rausche, das Original bleibt unverändert) und eins (das Original wird durch 100% Rauschen ersetzt) gewählt werden.

Parameter:
Pegel

Rauschpegel, muss immer als Fließkommazahl größer Null und kleiner oder gleich Eins angegeben werden.

Modus

Wert	Beschreibung
0	Rauschpegel als Prozentsatz der Amplitude eingeben, von 1 bis 100.
1	Rauschpegel in Dezibel eingeben, von -21 dB bis 0 dB.

5.19 normalize (Normalisierer)

Interner Name:

normalize

Modul-Typ:

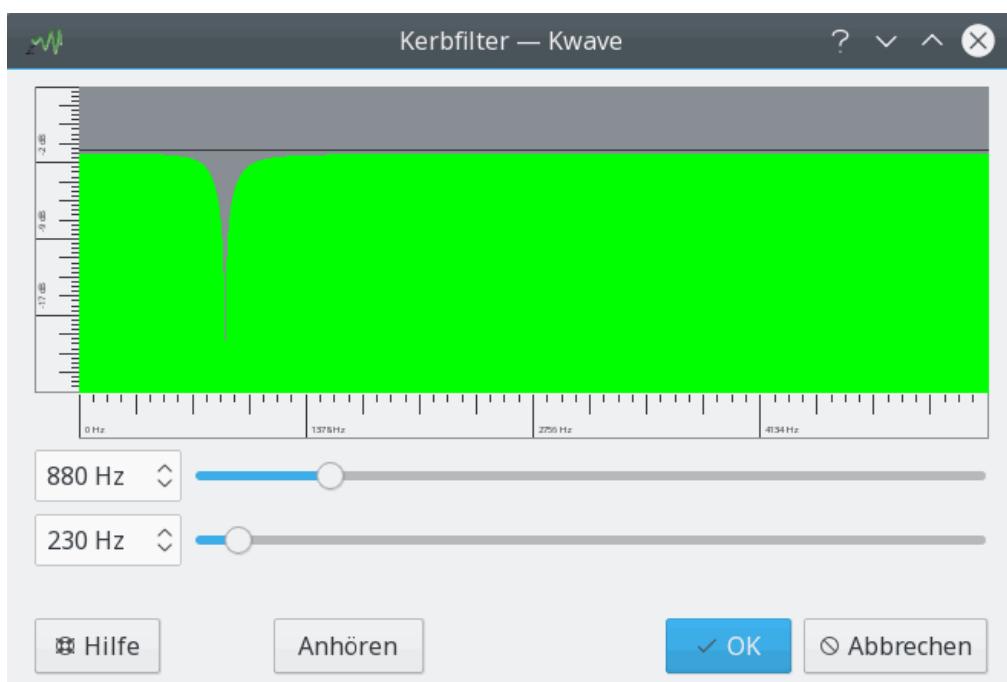
Effekt

Beschreibung:

Normalisiert den Lautstärke-Pegel der aktuellen Auswahl. Verwenden Sie dies wenn Ihre Datei zu leise oder zu laut klingt.

Der Algorithmus wurde vom *normalize*-Projekt übernommen und stammt ursprünglich von [Chris Vaill](#).

5.20 notch_filter (Kerbfilter)



Interner Name:

notch_filter

Modul-Typ:

Effekt

Beschreibung:

Wendet einen Kerbfilter auf die aktuelle Auswahl an. Ein Kerbfilter entfernt einen kleinen Frequenzbereich um eine *Mittenfrequenz* und lässt alle Frequenzen die mehr als die Hälfte der *Bandbreite* unterhalb oder oberhalb der Mittenfrequenz liegen passieren.

Verwenden Sie diesen Filter um einzelne Störfrequenzen herauszufiltern.

Der Filter ist zweiten Grades und basiert auf der Implementierung von [Juhana Sadeharju](#).

Parameter:

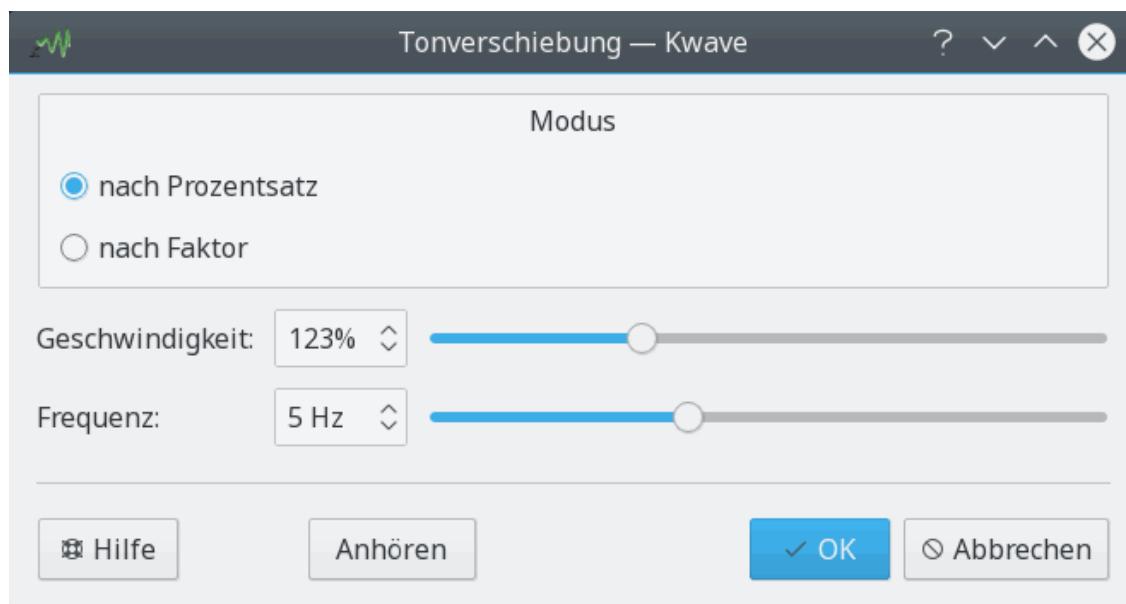
Frequenz

Mittenfrequenz des Filters in Hz, muss unterhalb der halben Abtastrate der Datei liegen.

Bandbreite

Bandbreite des Filters in Hz.

5.21 pitch_shift (Tonverschiebung)



Interner Name:

pitch_shift

Modul-Typ:

Effekt

Beschreibung:

Der Tonverschiebungs-Effekt ändert das Ausgangssignal indem er die Geschwindigkeit des Inhalts ändert, jedoch unter Beibehaltung der Länge. Sie können eine relative Geschwindigkeit angeben, entweder als Faktor zwischen 1/10 und x5, oder als Prozentsatz zwischen 1% und 400% der Original-Geschwindigkeit.

Ein Geschwindigkeitsfaktor unter 1,0 verschiebt die Tonhöhe nach unten (tiefere Stimmen, lässt Stimmen älter klingen), der Faktor 1,0 hat keine Auswirkung und ein Faktor über 1,0 verschiebt die Tonhöhe nach oben (höhere Stimmen, Mickey-Mouse-Effekt).

Die Implementierung basiert auf der Arbeit von [Jeff Tranter](#) und [Stefan Westerfeld](#)

Parameter:

Geschwindigkeit

Faktor für die Geschwindigkeitsänderung, muss als Fließkommazahl zwischen 0,001 und 4,0 angegeben werden.

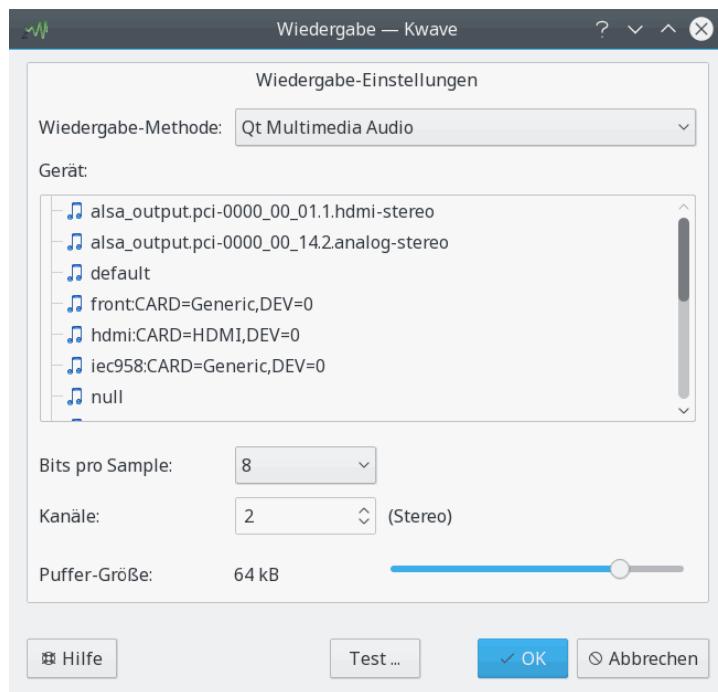
Frequenz

Frequenz in Hz die intern vom Filter verwendet wird, muss zwischen 2,0 und 10,0 liegen.

Modus

Wert	Beschreibung
0	Geschwindigkeits-Faktor als Faktor von 1/10 bis x5 eingeben.
1	Geschwindigkeits-Faktor als Prozentsatz von 1 bis 400 eingeben.

5.22 playback (Wiedergabe)



Interner Name:

playback

Modul-Typ:

Funktion

Beschreibung:

Zeigt ein Dialogfenster um die Parameter der Wiedergabe einzustellen. Bitte lesen Sie im Abschnitt in diesem Handbuch für weitere Informationen.

Parameter:

Wiedergabemethode

Wiedergabe-Methode, siehe PlayBackParam.h.

Wiedergabegerät

Eine Zeichenfolge die das Ausgabegerät oder dessen Kanal bestimmt. Die Bedeutung hängt von der gewählten Wiedergabemethode ab.

Kanäle

Anzahl der Spuren die für die Wiedergabe verwendet werden sollen, momentan werden nur 1 (Mono) und 2 (Stereo) unterstützt.

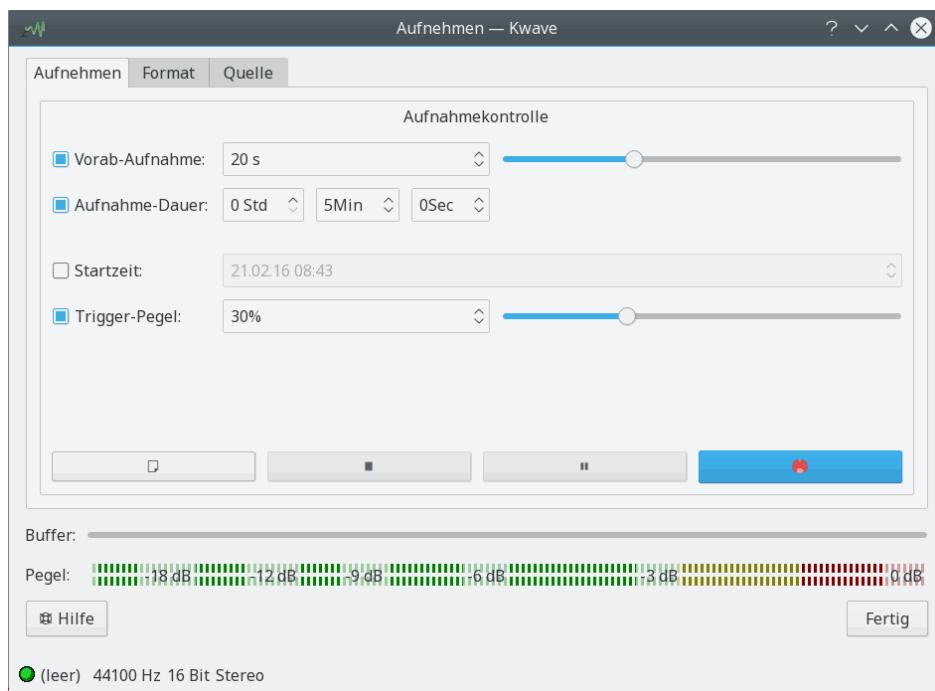
Bits pro Sample

Anzahl der Bits pro Sample, erlaubte Werte sind 8, 16, 24 und 32, je nach Wiedergabe-Methode und Wiedergabe-Gerät.

Puffer-Größe

Legt den Exponenten für die Berechnung der Größe des Wiedergabe-Puffers fest, als 2^n , z. B. ergibt der Wert 16 eine Puffergröße von $2^{16} = 64$ kB.

5.23 record (Aufnahme)



Interner Name:

record

Modul-Typ:

Funktion

Beschreibung:

Zeigt ein Dialogfenster um die Parameter der Aufnahme einzustellen und um eine Aufnahme durchzuführen. Bitte lesen Sie im Abschnitt in diesem Handbuch für weitere Informationen.

Parameter:

Aufnahmemethode

Aufnahme-Methode, siehe RecordParams.h.

Vorab-Aufnahme aktiviert

Vorab-Aufnahme aktivieren/deaktivieren (1 wenn aktiviert, 0 wenn deaktiviert).

Vorab-Aufnahme-Dauer

Anzahl der Sekunden für die Vorab-Aufnahme.

Aufnahmedauer begrenzen

Aktivieren/Deaktivieren der Begrenzung der Aufnahmedauer (1 für begrenzt, 0 für unbegrenzt).

Aufnahmedauer

Dauer der Aufnahme in Sekunden.

Startzeit verwenden

Aktivieren/Deaktivieren der Startzeit (1 wenn verwendet, 0 wenn nicht verwendet).

Startzeit

Datum/Uhrzeit mit dem Startzeitpunkt, in ISO-Format.

Trigger-Pegel verwenden

Aktivieren/Deaktivieren des Trigger-Pegels (1 für aktiviert, 0 für deaktiviert)

Trigger-Pegel

Trigger-Pegel in Prozent.

Aufnahmegerät

Ein String der das Aufnahmegerät identifiziert.

Kanäle

Anzahl der Spuren der Aufnahme.

Abtastrate

Abtastrate in Samples pro Sekunde.

Kompression

zum Speichern der Samples verwendete Kompression

Sample-Format

Zum Speichern der Samples verwendetes Format , siehe [Abschnitt über die Sample-Formate](#).

Bits pro Sample

Anzahl der Bits pro Sample, sollte 8 , 16, 24 oder 32 sein.

Anzahl der Puffer

Bestimmt die Anzahl der zu verwendenden Puffer.

Puffer-Grösse

Legt den Exponenten für die Berechnung der Größe der Aufnahme-Puffer fest, als 2^n , z. B. ergibt der Wert 16 eine Puffergröße von $2^{16} = 64$ kB.

Alternative Parameter:

Aufnahme-Modul Direktauswahl

Kann als einzelner Parameter verwendet werden um die Aufnahme einzurichten. Die folgenden Werte sind erlaubt:

Wert	Beschreibung
format	Öffnet das Aufnahme-Dialogfenster und wählt die Karteikarte Format .
source	Öffnet das Aufnahme-Dialogfenster und wählt die Karteikarte Quelle .
start_now	Öffnet das Aufnahme-Dialogfenster und startet direkt die Aufnahme.

5.24 reverse (Umkehren)

Interner Name:

reverse

Modul-Typ:

Effekt

Beschreibung:

Dieser einfache Effekt kehrt den Inhalt der aktuellen Auswahl um.

5.25 samplerate (Abtastrate ändern)

Interner Name:

samplerate

Modul-Typ:

Effekt

Beschreibung:

Ändert die Abtastrate der aktuellen Auswahl oder der gesamten Datei.

Parameter:

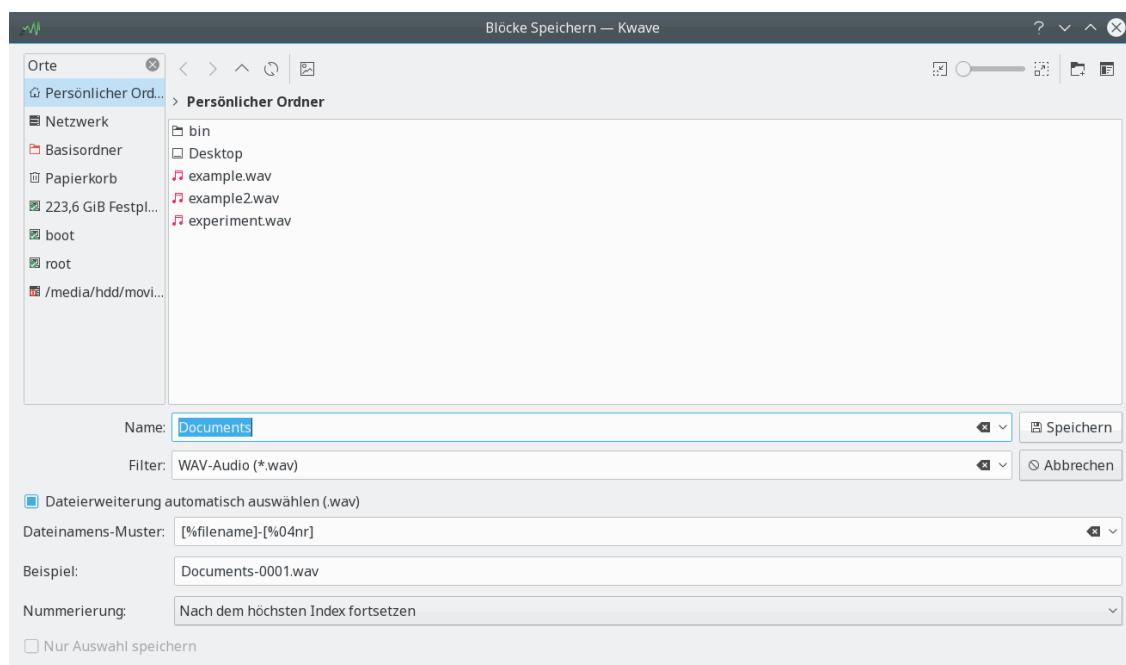
neue Abtastrate

die neue Abtastrate in Samples pro Sekunde (Fließkommazahl)

Modus (optional)

Wenn dieser Parameter verwendet und auf den Wert "all" gesetzt wurde, wirkt sich der Effekt auf die gesamte Datei aus. Andernfalls wird er nur auf die aktuelle Auswahl angewendet.

5.26 saveblocks (Blöcke speichern)



Interner Name:

saveblocks

Modul-Typ:

Funktion

Beschreibung:

Speichert alle Bereiche zwischen Markierungen, jeden in eine eigene Datei. Der Name der einzelnen Dateien kann über ein Muster angepasst werden das den Original-Dateinamen, einen Index und die Anzahl der Abschnitte enthalten kann.

Es ist auch zulässig im Dateinamens-Muster Schrägstriche als Pfad-Trennzeichen zu verwenden, wodurch es möglich wird die Abschnitte in verschiedene Unterordner zu speichern. Bitte beachten Sie dass Leerzeichen vor und nach diesen Pfad-Trennzeichen stillschweigend entfernt werden um zu verhindern dass Ordner-Namen erzeugt werden die mit Leerzeichen beginnen oder enden.

Parameter:

Name

Name der Original-Datei, wird als Basis für die Dateinamen verwendet.

Muster

Ein Muster das verwendet wird um die Namen der Dateien zu bestimmen. Es können die folgenden Platzhalter verwenden die beim Erzeugen der endgültigen Dateinamen ersetzt werden:

Platzhalter	Beschreibung
[%nr]	Wird durch den Index der jeweiligen zu speichernden Datei ersetzt.
[%count]	Wird durch die Anzahl der zu speichernden Abschnitte ersetzt.
[%total]	Wird durch den Index der letzten zu speichernden Datei ersetzt.
[%filename]	Wird durch den Basis-Dateinamen, ohne Pfad und ohne Erweiterung ersetzt.
[%fileinfo{Schlüsselwort}]	Wird mit dem Inhalt einer Datei-Information ersetzt, die über das angegebene <i>Schlüsselwort</i> ausgewählt wird. Siehe Abschnitt für eine Liste aller verfügbaren Schlüsselwörter.
[%title]	Wird mit dem Titel des Blocks ersetzt, der aus dem Beschreibungstext der Markierung am <i>Anfang</i> des Blocks entnommen wird. Falls dieser Text fehlt, wird auf den Titel der Datei ausgewichen (siehe Datei-Information Name). Sollte dieser ebenfalls fehlen, wird auf den oben beschriebenen Basis-Dateinamen ausgewichen.

Alle numerischen Platzhalter können nach dem "%" eine Zahl enthalten die die Anzahl der Stellen festlegt. Wenn der Zahl eine 0 vorangestellt wird, wird das Ergebnis führende Nullen enthalten, andernfalls wird es führende Leerzeichen enthalten.

Beispiel: [%04nr] erzeugt eine Zahl zwischen 0001 und 9999.

Nummerierungs-Modus

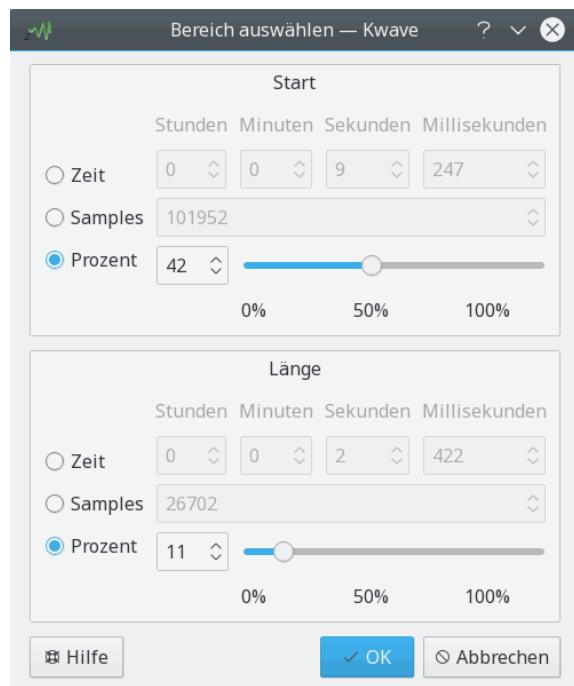
Bestimmt wo die Nummerierung beginnen soll.

Wert	Beschreibung
0	Nach dem höchsten gefundenen Index fortsetzen, dies verhindert dass existierende Dateien überschrieben werden.
1	Den Index immer mit 1 starten, mit dem Risiko dass existierende Dateien überschrieben werden.

nur die Auswahl

Wert	Beschreibung
0	Alle Abschnitte der gesamten Datei speichern.
1	Speichert nur die Abschnitte die in der aktuellen Auswahl liegen. Wenn nichts ausgewählt ist wird die ganze Datei gespeichert.

5.27 selectrange (Bereich auswählen)



Interner Name:

selectrange

Modul-Typ:

Funktion

Beschreibung:

Zeigt ein Dialogfenster einen Bereich an Samples auszuwählen. Der Start und die Länge des Bereichs können als Zeit in Millisekunden, als Position in Samples oder als Prozentsatz der aktuellen Länge der Datei angegeben werden.

Parameter:

Start-Modus

Bestimmt die Einheit in der der *Start* der Auswahl angegeben wird.

Wert	Beschreibung
0	Millisekunden
1	Samples
2	Prozentsatz der Länge der Datei

Bereichs-Einstellung

Bestimmt die Einheiten in denen die *Länge* der Auswahl angegeben wird. Siehe Parameter *Start-Modus* für eine Liste erlaubter Werte.

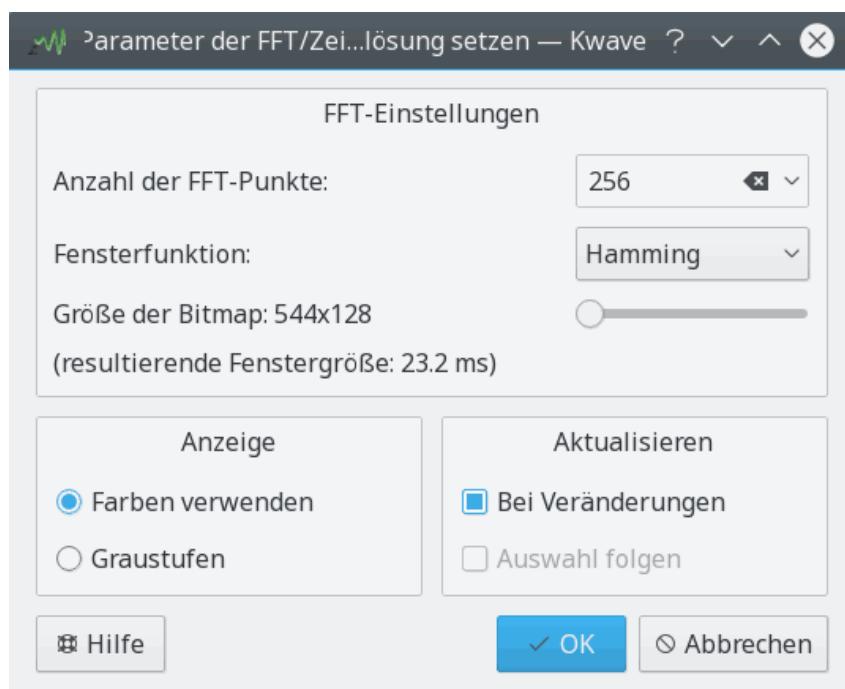
Start

Start der Auswahl, in Millisekunden, Samples oder Prozentsatz der Länge der Datei, in Abhängigkeit des Parameters *Bereichs-Modus*.

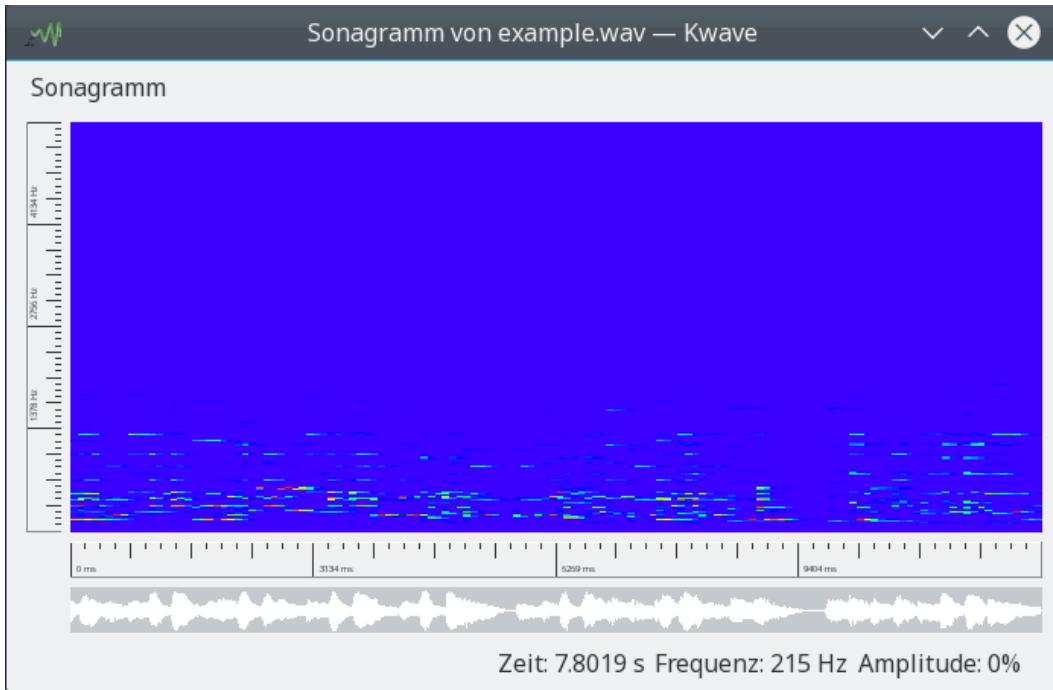
Länge

Länge der Auswahl, in Millisekunden, Samples oder Prozentsatz der Länge der Datei, in Abhängigkeit des Parameters *Bereichs-Modus*.

5.28 sonagram (Sonogram)



Das Kwave Handbuch



Interner Name:
sonagram

Modul-Typ:
Funktion

Beschreibung:

Wertet den Inhalt der aktuellen Auswahl mit Hilfe eines *Sonograms* aus. Ein Sonogramm ist eine Auswertung eines Signals über die Zeit (X-Achse), die Frequenz (Y-Achse) und die Intensität (Farbe).

Parameter:

FFT-Punkte

Anzahl der FFT-Punkte, eine ganze Zahl zwischen 4 und 32767 die die Auflösung im Frequenzbereich bestimmt.

Fensterfunktion

Die für die FFT-Berechnung verwendete Fensterfunktion, folgende Werte werden unterstützt:

Wert	Beschreibung
none	keine Fensterfunktion
hamming	Hamming-Fenster
hanning	Hanning-Fenster
blackman	Blackman-Fenster
Dreieck	Dreieck-Fenster

Farben verwenden

Wenn ein Wert ungleich Null übergeben wurde wird eine Farbdarstellung für die Intensität verwendet, bei Null werden Graustufen verwendet.

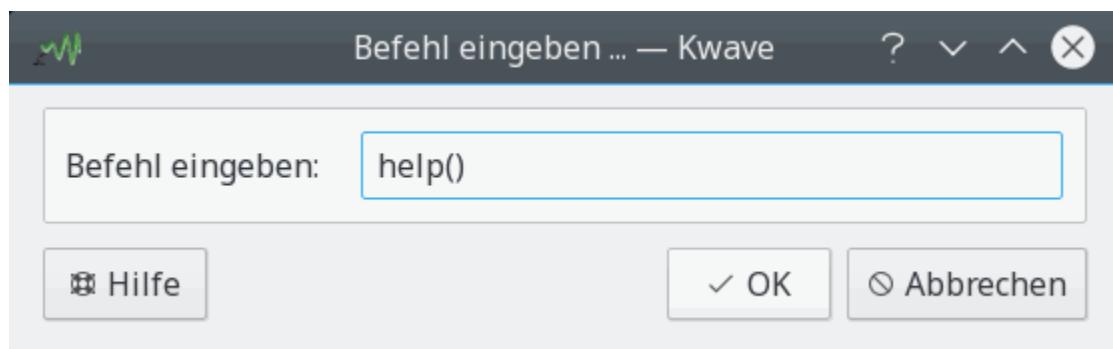
Änderungen verfolgen

Wenn eine Wert ungleich Null übergeben wird, wird das Sonogramm aktualisiert wenn sich der Inhalt des ausgewerteten Bereichs geändert hat. Beim Wert Null wird niemals aktualisiert.

der Auswahl folgen

Noch nicht implementiert, übergeben Sie Null für diesen Parameter.

5.29 stringenter (Befehl eingeben)



Interner Name:

stringenter

Modul-Typ:

Funktion

Beschreibung:

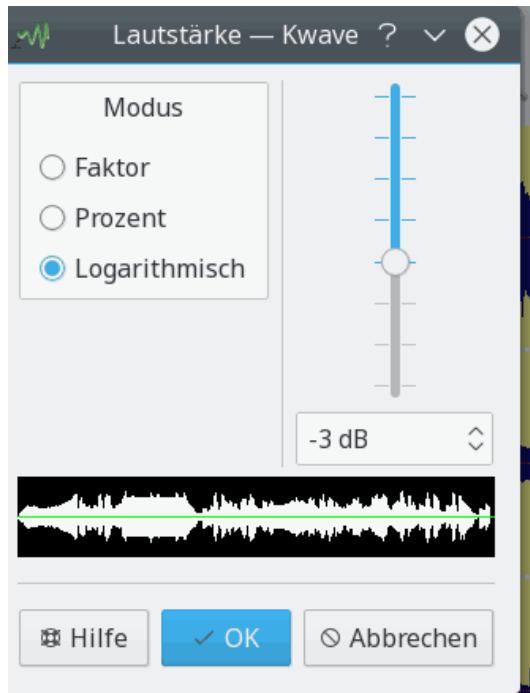
Ein kleines Dialogfenster in dem man einen Kwave Textbefehl eingeben kann. Für eine Beschreibung lesen Sie bitte das Kapitel in diesem Handbuch.

Parameter:

Vorgabe (optional)

Ein Text der im Eingabefeld des Dialogs angezeigt wird. Dieser Parameter ist optional, wenn er weggelassen wird ist das Eingabefeld anfangs leer.

5.30 volume (Lautstärke)



Interner Name:

volume

Modul-Typ:

Effekt

Beschreibung:

Mit diesem Modul können Sie die Lautstärke der aktuellen Auswahl um einen konstanten Faktor ändern. Im zugehörigen Dialogfenster kann man diesen Faktor als *numerischen Faktor* als Fließkommazahl zwischen 0,10 und 10,0, als *Prozentsatz* zwischen 1 und 1000, oder in *Dezibel* zwischen -21 und +21 angeben.

Verwenden Sie einen Faktor über 1,0 (oder Prozentsatz über 100 oder mehr als 0 dB) wenn die Datei zu leise ist, oder einen Faktor unter 1,0 (Prozentsatz unter 100 oder weniger als 0 dB) wenn die Datei zu laut ist.

Parameter:

Faktor

Eine Fließkommazahl mit dem Verstärkungsfaktor.

Modus

Wert	Beschreibung
0	Faktor
1	Prozentsatz
1	Dezibel

5.31 zero (Null-Generator)

Interner Name:

zero

Modul-Typ:

Effekt

Beschreibung:

Dieses Modul kann auf zwei verschiedene Arten verwendet werden. Wird es ohne Parameter aufgerufen, löscht es den aktuell ausgewählten Bereich durch Überschreiben mit Nullen. Wenn es mit zwei Parametern aufgerufen wird, fügt es am Anfang der aktuellen Auswahl Stille mit angegebener Länge ein.

Parameter:

Modus der Längenangabe

Bestimmt die Einheit in der die *Länge* der eingefügten Stille angegeben wird.

Wert	Beschreibung
0	Millisekunden
1	Samples
2	Prozentsatz der Länge der Datei

Länge

Länge der Stille die eingefügt werden soll, in Millisekunden, Samples oder Prozentsatz der Dateilänge, abhängig vom Parameter *Modus der Längenangabe*.

Kapitel 6

Fragen und Antworten

1. Was brauche ich um Kwave zu kompilieren?

Bitte lesen Sie hierzu die [Entwickler-Dokumentation](#).

2. Welche Soundkarten unterstützt Kwave?

Kwave braucht keine Unterstützung für irgendeine besondere Soundkarte. Die Soundkarte muss nur von Ihrem Betriebssystem unterstützt werden und Kwave nutzt dann sein Interface zum Betriebssystem-Soundkarten-Treiber mit Hilfe von OSS oder ALSA.

3. Warum verbraucht Kwave mehr Speicher als es die Grösse der geöffneten Datei erwarten lässt?

Der Grund dafür liegt im internen Speichersystem von Kwave, in dem alle Samples in 32-bit Ganzzahlen (Integers) gespeichert werden. Dies war einfach zu programmieren, machte die Anwendung schneller und ein bisschen betriebssicherer. Deshalb werden 8-Bit Dateien mit rund einem Megabyte um die vier Megabyte im Speicher belegen. Vielleicht werden wir das irgendwann in der Zukunft ändern...

4. Welche Soundformate unterstützt Kwave?

Kwave unterstützt zur Zeit .wav Dateien mit 8, 16 und 24 bits pro Sample, mit beliebiger Anzahl von Spuren (einschließlich Mono und Stereo natürlich). Zusätzlich unterstützt es alle Dateiformate von libaudiofile sowie andere wie Ogg/Vorbis und MP3.

5. Was ist, wenn ich Dateien habe die Kwave nicht unterstützt?

Wenn Sie mit einem anderen Format arbeiten müssen, dann können Sie dieses in das .wav Format konvertieren. Eine gute Auswahl an Werkzeugen bietet das [SoX](#) Paket, das auch umfangreiche Dokumentationen enthält!

6. Ich bekomme Fehler wenn ich versuche die Wiedergabe zu starten?

Vielleicht haben Sie eine Kombination von Wiedergaberate und Sample-Grösse gewählt, die nicht von ihrem Sound-Treiber und / oder ihrer Soundkarte unterstützt wird. Versuchen Sie zuerst Playback mit 8 bits pro Sample und Mono, dies sollte immer funktionieren. Dann versuchen Sie die Bits pro Sample und die Stereo-Wiedergabe schrittweise zu erhöhen. Beachten Sie dabei aber, dass einige Wiedergaberäte nicht vollständig von einigen Soundkarten unterstützt werden.

7. Die Wiedergabe scheint etwas zu tun, aber ich höre nichts?

Vielleicht haben Sie vergessen die Lautstärke des Wiedergabe-Kanals zu erhöhen. Für das Ändern der Wiedergabelautstärke ist Kwave nicht verantwortlich.

8. Einige Dateien werden mit halber Geschwindigkeit wiedergegeben?

Versuchen Sie ein anderes Gerät zur Soundwiedergabe auszuwählen.

Das Kwave Handbuch

9. *Die Wiedergabe wird manchmal gestört und unterbrochen?*

Sie sollten die Grösse des Wiedergabepuffers erhöhen um eine "glattere" Wiedergabe zu bekommen (dies führt auch zu einer langsameren Reaktion der Wiedergabe- Kontrolle).

10. *Die Wiedergabe hält nicht an, wenn ich sofort die Stop-Taste drücke?*

Der Grund dafür ist, dass der Soundtreiber, in dem Moment in dem Sie die Stop-Taste drücken, bereits Wiedergabedaten von Kwave erhalten hat. Verringern Sie die Grösse des Wiedergabe Puffers und die Reaktion sollte schneller einsetzen (dabei werden aber Aussetzer wahrscheinlicher).

11. *Wird ALSA unterstützt?*

Ja, seit v0.7.4 für Aufnahme und Wiedergabe

12. *Wie steht es mit der Wiedergabe von 18, 20, 24 oder 32 bits pro Sample oder mehr als zwei Spuren?*

Dies ist seit v0.7.4 möglich mit Hilfe von ALSA.

Kapitel 7

Danksagungen und Lizenz

Kwave

Programm Copyright von 1998-2000 Martin Wilz martin@mwilz.de

Dokumentation copyright (C) seit 2000 Thomas Eschenbacher thomas.eschenbacher@gmx.de

Für eine vollständige Liste der Autoren und Lizenzen lesen Sie bitte die Datei [LICENSES](#), die in der Quellcode-Distribution enthalten ist. Weiterhin gibt es eine Datei namens [AUTHORS](#), die alle Autoren von Kwave auflistet;

Dokumentation Copyright (C) 2020 Thomas Eschenbacher thomas.eschenbacher@gmx.de

Übersetzung Sven-Steffen Arndt ssa29@gmx.de und Thomas Eschenbacher Thomas.Eschenbacher@gmx.de

Diese Dokumentation ist unter den Bedingungen der [GNU Free Documentation License](#) veröffentlicht.

Dieses Programm ist unter den Bedingungen der [GNU General Public License](#) veröffentlicht.

7.1 Haupt-Autoren

- **Martin Wilz** martin@wilz.de
Urheber des Projekts, Entwicklung von 1998-2000
- **Thomas Eschenbacher** thomas.eschenbacher@gmx.de
Leiter des Projekts seit 2000, Kern-Entwicklung

7.2 Haupt-Beitragende

- **Aurelien Jarno** aurel32@debian.org
[debian](#) Pakete, Patches
- **Carlos R** pureacetone@gmail.com
Spanische Übersetzung
- **David Flogeras** dflogera@nbnet.nb.ca
Kerbfilter-Modul

- **Gilles Caulier** caulier.gilles@free.fr
Französische Übersetzungen, Startbild, Beta-Tester
- **Pavel Fric** pavelfric@seznam.cz
Tschechische Übersetzung
- **Ralf Waspe** rwaspe@web.de
Hilfe/Über-Modul
- **Sven-Steffen Arndt** ssa29@gmx.de
Homepage, deutsche Übersetzung

7.3 Weitere Beitragende, Copyright-Halter und andere

- **Aaron Holtzman** aholtzma@ess.engr.uvic.ca
libkwave/cpu_accel.cpp
- **Bertrand Songis** bsongis@gmail.com [*historisch*]
Fehlerbehebungen in der französischen Übersetzung, Ersatz von patentiertem Code in libaudiofile, debian bug 419124
- **Carsten Lohrke** carlo@gentoo.org
svn r2163, Patch zur Detektion von libaudiofile
- **Chris Vaill** chrisvaill@gmail
Codebasis des Normalisierer-Moduls
- **David Faure** faure@kde.org
cmake/FindAlsa.cmake
- **Diederick de Vries** diederick76@gmail.com
Pakete für Crux-Linux
- **Espen Sand** espen@kde.org + **Mirko Boehm** mirko@kde.org
K3AboutContainer, Grundlage von KwaveAboutContainer
- **Everaldo Coelho** contact@everaldo.com
das crystal Icon-Thema <http://www.everaldo.com/crystal/>
- **Jaroslav Kysela**
Teile von plugins/playback/PlayBack-ALSA.cpp
- **Jeff Tranter**
Teile von plugins/pitch_shift/PitchShiftFilter.{h,cpp}
- **Juhana Sadeharju** kouhia@nic.funet.fi
plugins/band_pass/BandPass.{h,cpp}, plugins/lowpass/LowPassFilter.cpp, plugins/notch_filter/NotchFilter.{h,cpp}
- **Kurt Roeck** Q@ping.be
svn r1370, Fehlerbehebung für Debian-Bug#288781, Generierung für amd64
- **Mark Donohoe (KDE)** donohoe@kde.org
einige Icons und Bilder für Werkzeugeiste und GUI

Das Kwave Handbuch

- **Martin Hinsch** vidas@sourceforge.net
Matrix-Klasse
- **Matthias Kretz** kretz@kde.org
cmake/FindAlsa.cmake
- **Miguel Freitas**
Teile von libkwave/memcpy.c
- **Richard Laerkaeng**, richard@goteborg.utfors.se
cmake/FindOggVorbis.cmake
- **Rik Hemsley** rik@kde.org
Aussteuerungsanzeige
- **Stefan Westerfeld** stefan@space.twc.de
Teile von plugins/pitch_shift/PitchShiftFilter.{h,cpp}
- **Joerg-Christian Boehme** joerg@chaosdorf.de
plugins/record/Record-PulseAudio.cpp plugins/record/Record-PulseAudio.h
- **Sebastian Trueg** trueg@k3b.org, **Gustavo Pichorim Boiko** gustavo.boiko@kdemail.net,
Michal Malek michalm@jabster.pl
Teile von plugins/export_k3b/K3BExportPlugin.cpp

7.4 Dank an

- **Martin Kuball** makube@user.sourceforge.net
Beta-Tester
- **Jorge Luis Arzola** arzolacub@gmx.de
Pakete für SuSE Linux
- **Michael Favreau** michel.favreau@free.fr
Pakete für Arch Linux
- **T.H.F. Klok and Cedric Tefft**
Betreuer der [id3lib](#) Bibliothek
- **Robert Leslie** rob@mars.org
Autor der [mad](#) mp3-Decoder-Bibliothek
- **Robert M. Stockmann** stock@stokkie.net
Pakete für Mandrake / X86_64
- **Erik de Castro Lopo** erikd@zip.com.au
Autor der [sndfile](#)-Bibliothek
- **Michael Pruett** mpruett@sgi.com
Autor der [audiofile](#) Bibliothek

Anhang A

Datei-Informationen

Schlüsselwort		Beschreibung
Album		Name des Albums wenn es sich um ein Album handelt das aus mehreren Medien besteht.
Annotation		Enthält allgemeine Kommentare über die Datei oder den Inhalt der Datei. Bei mehrzeiligen Kommentaren sollte jede Zeile mit einem Punkt enden. Keine Zeilenumbrüche einfügen!
Archival location		Gibt an wo der Inhalt der Datei archiviert wird.
Author		Benennt den Namen des Autors des ursprünglichen Werkes oder der Datei. Beispiel: „van Beethoven, Ludwig“
Lower Bitrate		Setzt die Untergrenze für die Bitrate in einem VBR-Bitstrom.
Bitrate Mode		Bitraten-Modus (ABR, VBR, CBR, usw.)
Bitrate		Nominale Bitrate des Audio-Stroms in Bits pro Sekunde
Upper Bitrate		Setzt die Obergrenze der Bitrate in einem VBR-Bitstrom.
Bits per Sample		Setzt die Anzahl der Bits pro Sample.
CD		Nummer der CD, wenn es sich um ein Album aus mehreren CDs handelt

Das Kwave Handbuch

CDS		Anzahl der CDs, wenn es sich um ein Album aus mehreren CDs handelt
Commissioned		Enthält den Namen der Person oder Organisation, die das Thema der Datei genehmigt hat.
Comments		Enthält allgemeine Kommentare über die Datei oder den Inhalt der Datei. Bei mehrzeiligen Kommentaren sollte jede Zeile mit einem Punkt enden. Keine Zeilenumbrüche einfügen!
Compression		Wählt die Komprimierungs-Methode für Audio-Daten zur Reduktion des Speicherplatz-Bedarfs.
Contact		Kontaktinformation für die Ersteller oder Distributoren des Stücks. Dies kann ein URL, eine E-Mail Adresse, oder eine Adresse des Hersteller-Labels sein.
Copyright		Hält die Copyright-Informationen der Datei fest. Sind mehrere Copyright-Angaben vorhanden, sind diese durch Semikolon und Leerzeichen zu trennen. Beispiel: „Copyright Linux-Gemeinschaft 2002“
Copyrighted		Gibt an ob der Inhalt der Datei urheberrechtlich geschützt ist oder nicht.
Date		Gibt das Datum an zu dem der ursprüngliche Titel erzeugt wurde. Beispiel: „2001-12-24“
Engineer		Nennt den Namen des Ton-Technikers der die Datei bearbeitet hat. Bei mehreren Technikern sind die Namen durch ein Semikolon und ein Leerzeichen zu trennen.
Estimated Length		Geschätzte Länge der Datei in Samples
Filename		Name der geöffneten Datei
File Size		Grösse der Datei in Bytes

Das Kwave Handbuch

Genre		Beschreibt das Genre bzw. den Stil des Originalwerkes. Beispiele: „Klassik“, „Pop“
ISRC		ISRC-Nummer des Stücks. Für weitere Informationen über ISRC-Nummern siehe die ISRC-Einführungs-Seite unter: http://isrc.ifpi.org/
Keywords		Enthält eine Liste von Schlüsselworten die sich auf die Datei oder deren Inhalt beziehen.
Labels		Die Liste aller Markierungen.
Length		Länge der Datei in Samples.
License		Lizenzinformation, z. B., „Alle Rechte Vorbehalten“, „Zur freien Verwendung“, eine URL zu einer Lizenz oder die Attribution-ShareAlike 4.0 International-Lizenz („verbreitet unter den Bedingungen der Attribution-ShareAlike 4.0 International Lizenz. Siehe https://creativecommons.org/licenses/by-sa/4.0/ für weitere Informationen“), usw.
Medium		Beschreibt den ursprünglichen Gegenstand des Stücks, bei der erstmaligen Aufnahme. Beispiel: „Orchester“
MIME-Typ		MIME-Typ des Dateiformates
Emphasis		Audio-Emphase Einstellung
Layer		MPEG Layer, I, II oder III
Mode Extension		MPEG Modus-Erweiterung (nur im Joint-Stereo Modus)
Version		MPEG Version, 1, 2 oder 2.5
Name		Enthält den Titel des Gegenstands der Datei. Beispiel: „Symphonie No.6, Op.68 „Pastoral““
Opus Frame Length		Opus Frame-Länge in ms (unterstützte Werte sind 2.5, 5, 10, 20, 40, und 60 ms)
Organization		Name der Organisation die das Stück produziert hat (z.B. der „Platten-Label“)

Das Kwave Handbuch

Original		Gibt an ob es sich beim Inhalt der Datei um ein Original handelt.
Performer		Der oder die Künstler, die das Werk ausführten. Bei klassischer Musik kann dies der Dirigent, das Orchester oder der Solist sein, oder der Vortragende bei einem Hörspiel.
Private		Gibt an ob der Inhalt der Datei nur für privaten Gebrauch ist.
Product		Bezeichnet den Namen oder den Titel des Produkts für den die Datei ursprünglich gedacht war. Beispiel: „Linux Audio-Sammlung“
Sample Format		Das zum Speichern der digitalisierten Audio-Samples verwendete Format. Beispiel: „32-Bit IEEE Fließkomma“
Sample Rate		Anzahl der Samples pro Sekunde
Software		Nennt den Namen des Software-Paketes das zur Erstellung der Datei verwendet wurde. Beispiel: „Kwave v0.6.4-1“
Source		Nennt den Namen der Person oder Organisation die das Original der Datei oder den Gegenstand zur Verfügung gestellt hat. Beispiel: „Chaotic Sound Research“
Source form		Gibt an von welchem Ursprungs-Medium das Stück digitalisiert wurde. Beispiele: „Langspielplatte-/Vinyl/90UPM“, „Audio-DAT“, „Kassette/CrO ₂ /60min“
Subject		Beschreibt den Inhalt der Datei. Beispiel: „Vogelgesang am frühen Morgen“
Technician		Nennt den Techniker der das Werk digitalisiert hat. Beispiel: „Torvalds, Linus“
Stück		Stück auf der CD wenn die Quelle eine CDROM war.

Das Kwave Handbuch

Stücke		Nummer des Stücks, wenn die Quelle eine CD war.
Channels		Setzt die Anzahl der Spuren des Signals.
Base Quality		Basisqualität der Kompression im VBR-Modus
Version		Kann verwendet werden um verschiedene Versionen des selben Stücks in einer Single-Sammlung zu unterscheiden. (z. B. Name des Remixes)

Tabelle A.1: Liste der Datei-Info-Schlüsselworte