

MASTER THESIS
Fiona Vogel

Bau eines hexaphonischen Tonabnehmers und Konvertierung analoger Töne von E-Gitarren in das MIDI-Format

FACHBEREICH TECHNIK
Faculty of Engineering

FAKULTÄT TECHNIK UND INFORMATIK
Department Informations- und Elektrotechnik
Faculty of Engineering and Computer Science
Department of Information and Electrical Engineering

Fiona Vogel

Bau eines hexaphonischen Tonabnehmers und Konvertierung analoger Töne von E-Gitarren in das MIDI-Format

Masterarbeit eingereicht im Rahmen der Masterprüfung
im gemeinsamen Masterstudiengang *Mikroelektronische Systeme*
am Fachbereich Technik
der Fachhochschule Westküste
und
am Department Informations- und Elektrotechnik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuer Prüfer: Prof. Dr.-Ing. Marc Hensel
Zweitgutachter: Prof. Dr.-Ing. Sönke Appel

Eingereicht am: 17.09.2024

Fiona Vogel

Thema der Arbeit

Bau eines hexaphonischen Tonabnehmers und Konvertierung analoger Töne von E-Gitarren in das MIDI-Format

Stichworte

Hexaphonischer Tonabnehmer für E-Gitarren, polyphone Tonhöhenerkennung, MIDI-Konvertierung

Kurzzusammenfassung

In dieser Arbeit wird ein hexaphonischer Tonabnehmer für E-Gitarren gebaut und eine Verarbeitungskette von Tonerzeugung bis Tonhöhenerkennung entwickelt. Die hexaphonisch getrennten Tonspuren werden in das branchenübliche MIDI-Format übersetzt. Dabei wird getestet, ob ein monophoner Algorithmus zur Tonhöhenerkennung mithilfe des hexaphonischen Tonabnehmers quasi-polyphon arbeiten kann.

Fiona Vogel

Title of Thesis

Construction of a hexaphonic pickup and conversion of analog electric guitar tones into the MIDI format

Keywords

Hexaphonic pickup for E-guitars, polyphonic pitch detection, MIDI conversion

Abstract

This thesis covers the construction of a hexaphonic pickup and the development of a processing chain from sound generation to pitch detection. The hexaphonically divided sound tracks are translated into the prevalent MIDI format. This method tests whether a monophonic pitch detection algorithm can work quasi-polyphonic while using a hexaphonic pickup.

Danksagung

Ich danke meinem Erstprüfer Prof. Dr.-Ing. Marc Hensel für die Bereitstellung dieses exotischen Themas und seinen beharrlichen Optimismus. Außerdem hat er liebenswürdigerweise grünes Filament verwendet, obwohl ich ihn nicht darum gebeten habe. Ein großer Dank gilt Henry Vogel, der alle verwendeten Gitarren-Samples eingespielt hat. Ansonsten hätte ich mein Credo, nur Bass-Gitarre zu spielen, aufgeben müssen. Zuletzt danke ich Christoph Rathjen für seine vollumfängliche Unterstützung während meines Master-Studiums, auch wenn ich dies aus niederen Beweggründen aufgenommen und uns somit für anderthalb Jahre Geld- und Zeiteinbußen beschert habe. Ich hoffe der Spaß, den er daran hatte, kann das kompensieren.

Inhaltsverzeichnis

Abbildungsverzeichnis	xiii
Tabellenverzeichnis	xvii
Abkürzungen	xix
Symbolverzeichnis	xxi
1 Einleitung	1
1.1 Motivation	1
1.2 Zielsetzung	1
2 Grundlagen	3
2.1 Aufbau einer Gitarre	3
2.2 Klangerzeugung	4
2.2.1 Akustisch	4
2.2.2 Elektrisch	5
2.3 Notation von Gitarrennoten	5
2.3.1 Notensystem	5
2.3.2 Tabulatur	5
2.4 Spieltechniken	6
2.5 Obertöne	8
2.6 Tonabnehmer	10
2.6.1 Funktionsweise	10
2.6.2 Bauformen	11
2.6.3 Resonanz	14
2.7 Passive Schaltung in einer E-Gitarre	18
2.8 Instrumentenkabel und Anschlusskontakte	20
2.9 Spannungspegel in der Tontechnik	21
2.9.1 Line-Pegel	21

2.9.2	Mikrofon-Pegel	22
2.9.3	Instrumenten-Pegel	22
2.9.4	Lautsprecher-Pegel	23
2.10	Hörbare Latenz beim Live-Spiel	23
2.11	Algorithmen zur Tonhöhenerkennung	23
2.12	MIDI-Format	24
2.12.1	Hardwareschnittstelle	24
2.12.2	Datenformat	25
2.12.3	Dateiformat	27
2.12.4	Minimalbeispiel für den Aufbau einer MIDI-Datei	31
3	Stand der Technik	33
3.1	Hexaphonische Tonabnehmer	33
3.2	MIDI-Tonabnehmer	35
3.3	MIDI-Interfaces	35
3.4	MIDI-Konverter	36
3.5	Pitch Detection Algorithmen	36
3.5.1	Ereignisse im Zeitbereich	36
3.5.2	Autokorrelation	37
3.5.3	Frequenzbereich	41
3.5.4	Maschinelles Lernen und neuronale Netze	44
4	Anforderungsanalyse	47
4.1	Systemaufbau	47
4.2	Systemkontext	48
4.3	Interessengruppen	49
4.4	Anwendungsfälle	51
4.4.1	Vorverarbeitung	51
4.4.2	MIDI-Konvertierung	52
4.5	Anforderungen	54
4.5.1	Hexaphonischer Tonabnehmer	54
4.5.2	Instrumentenkabel	55
4.5.3	Vorverarbeitung	56
4.5.4	MIDI-Konvertierung	56
4.5.5	Allgemein	58

5 Gesamtkonzept	61
5.1 Schnittstellen	61
5.2 Vorgehensweise	62
5.2.1 Beschaffung der E-Gitarre	62
5.2.2 Konstruktion eines hexaphonischen Tonabnehmers	63
5.2.3 Beschaffung der Vorverarbeitung	63
5.2.4 Beschaffung des Gitarrenkabels	65
5.2.5 Zugriff und Visualisierung der digitalisierten Daten	65
5.2.6 Auswahl eines PDA zur Tonanalyse	65
5.2.7 Tonanalyse	66
5.2.8 Verwendung von Audiodatenbanken	66
5.2.9 MIDI-Konvertierung	67
6 Hexaphonischer Tonabnehmer	69
6.1 Konzept	69
6.1.1 Aufbau des Gehäuses	69
6.1.2 Auswahl der Magneten	70
6.1.3 Auswahl des Spulendrahts	70
6.1.4 Variation der Wicklungszahl	70
6.1.5 Messung des Spulenwiderstands	72
6.1.6 Herstellen der elektrischen Verbindung	72
6.1.7 Montage in der Gitarre	73
6.1.8 Messung des Amplitudengangs	73
6.2 Umsetzung	76
6.2.1 Gehäusedruck	76
6.2.2 Wicklung	76
6.2.3 Wachsen	78
6.2.4 Messung des Spulenwiderstandes	79
6.2.5 Herstellen der elektrischen Verbindung	79
6.2.6 Montage in der Gitarre	80
6.2.7 Kontaktweg vom Tonabnehmer zum Elektronikfach	81
6.2.8 Messung des Amplitudengangs	82
6.3 Weiterentwicklung des betreuenden Erstprüfers	86

7 Instrumentenkabel und Anschlusskontakte	91
7.1 Konzept	91
7.1.1 Aufbau des Kabels und des Übertragungsweges	91
7.1.2 Auswahl des Kabelmaterials	91
7.1.3 Aufbau der Übergabeeinheit	92
7.2 Umsetzung	93
7.2.1 Instrumentenkabel	93
7.2.2 Aufbau der Übergabeeinheit	93
8 Tonhöhenanalyse und MIDI-Konvertierung	95
8.1 Aufzeichnung der Gitarrentöne	95
8.2 Tonhöhenanalyse	97
8.2.1 Aufbau des Programms	97
8.2.2 Übergabeparameter	99
8.2.3 Ergänzung der Amplitudenbestimmung	100
8.2.4 Analyse statischer Töne	100
8.2.5 Analyse mehrerer Töne auf einer Saite	105
8.2.6 Interferenz zwischen zwei Tonabnehmerspulen	109
8.2.7 Analyse mehrere Töne auf allen Saiten	111
8.2.8 Analyse von Intervallen auf der E- und A-Saite	112
8.2.9 Zusammenfassung der Analyse-Ergebnisse	117
8.3 MIDI-Konvertierung	119
8.3.1 Zuordnung der ermittelten Frequenzen zu den Tonspuren	120
8.3.2 Konvertierung in MIDI-Angaben	121
8.3.3 Übersetzung in Bytes und Datei-Erstellung	123
8.3.4 Öffnen der erstellten MIDI-Datei im MIDI-Editor	126
8.4 Latenz	129
8.4.1 YIN	129
8.4.2 YIN und MIDI-Konvertierung	131
9 Test	137
9.1 Tonabnehmer	137
9.2 Instrumentenkabel	138
9.3 Vorverarbeitung	139
9.4 MIDI-Konvertierung	140
9.5 Allgemein	142

10 Fazit und Ausblick	145
10.1 Hexaphonischer Tonabnehmer	145
10.2 MIDI-Konvertierung	146
Literaturverzeichnis	149
A Anhang	155
A.1 Frequenzen der gleichstufigen Stimmung und ihre Tonnamen und MIDI-Nummern	155
A.2 Programmablaufdiagramme	161
A.3 Weitere Ergebnisse der Funktion <i>yin</i>	167
Stichwortverzeichnis	167
Selbstständigkeitserklärung	175

Abbildungsverzeichnis

2.1	Wesentliche Bestandteile einer Gitarre	4
2.2	Gitarrennoten im Notensystem notiert	5
2.3	Gitarrennoten in einer Tabulatur notiert	6
2.4	Gitarrennoten mit eingetragenen Spieltechniken	6
2.5	Idealer räumlicher Verlauf der Saitenschwingungen über eine Mensur von 628 mm	9
2.6	Ideales Linienspektrum für zehn Obertöne	10
2.7	Einfacher Aufbau eines Single Coil-Tonabnehmers	12
2.8	Aufbau eines Humbucker-Tonabnehmers	13
2.9	Induzierte Spannungen am Humbucker-Tonabnehmer nach [29]	13
2.10	Aufbau eines Piezo-Stegtonabnehmers	14
2.11	Ersatzschaltbild eines Tonabnehmers	15
2.12	Variation von R_1 bei $C_1 = 150 \text{ pF}$ und $L_1 = 3.5 \text{ H}$	16
2.13	Variation von C_1 bei $R_1 = 10 \text{ k}\Omega$ und $L_1 = 3.5 \text{ H}$	17
2.14	Variation von L_1 bei $R_1 = 10 \text{ k}\Omega$ und $C_1 = 150 \text{ pF}$	18
2.15	Ersatzschaltbild eines Tonabnehmers mit externer Beschaltung	18
2.16	Ersatzschaltbild eines Tonabnehmers mit vereinfachter externer Beschaltung	19
2.17	Variation von R_3 bei $R_1 = 10 \text{ k}\Omega$, $L_1 = 3.5 \text{ H}$ und $C = 800 \text{ pF}$	19
2.18	Passive Schaltung der E-Gitarre aus Kapitel 5.2.1	20
2.19	MIDI Standard Hardware [47]	25
2.20	Beispiel-Tonabfolge	31
3.1	Einfluss von Obertönen auf die Anzahl der Nulldurchgänge	37
3.2	Verlauf der Autokorrelation einer Sinusschwingung	38
3.3	Schritte der Cepstrum-Alanalyse [16]	43
4.1	Systemaufbau	48
4.2	Systemkontext	49
4.3	Anwendungsfälle der Vorverarbeitung	51

4.4 Anwendungsfälle der MIDI-Konvertierung	53
6.1 3D-Modell eines Gehäuses für hexaphonische Tonabnehmer im Single Coil-Format	70
6.2 Berechnung der Radien der Drahtwicklungen	71
6.3 Verlauf von $i(t)$ und $u_{ind}(t)$	74
6.4 Verlauf von $i(t)/f$ und $u_{ind}(t)/f$	75
6.5 Verlauf von $H'(f)$, $H'(f) \cdot u_{ind}(t)$ und $H'(f) \cdot u_{ind}(t)/f$	76
6.6 Aufbau zum Wickeln der Spulen	77
6.7 Zusammengesetzter hexaphonischer Tonabnehmer nach dem Wachsen	79
6.8 Hexaphonischer Tonabnehmer und Anschlussplatine auf dem Montageblech	81
6.9 Ausgebauter Humbucker-Tonabnehmer	81
6.10 Eingebauter hexaphonischer Tonabnehmer	82
6.11 Messspule mit Anschlussplatine	84
6.12 Gemessene Amplitudengänge des Tonabnehmers	85
6.13 Wickelmaschine des betreuenden Erstprüfers	87
6.14 Python-Programmfenster zur Steuerung des Schrittmotors	88
6.15 Tonabnehmerspulen im Wachsbad	88
6.16 Bestandteile des Tonabnehmers des betreuenden Erstprüfers	89
7.1 Entwurf der Übergabeeinheit	92
7.2 Kontaktwege innerhalb der Übergabeeinheit	93
7.3 Übergabeeinheit an der Gitarre angeschlossen	94
8.1 Einstellungen am Audiointerface für die Eingänge 1 und 2	95
8.2 Aufgenommene Signale der E- und A-Saite für einen Ton	96
8.3 Beispielausgabe der Funktion yin	98
8.4 Ausgabe der Funktion yin ergänzt um die Signalamplitude	101
8.5 Ausgabe der Funktion yin für den Ton Ais auf der E-Saite	103
8.6 Ausgabe der Funktion yin für den Ton d auf der E-Saite	104
8.7 Ausgabe der Funktion yin für den Ton d' auf der E-Saite	105
8.8 Ausgabe der Funktion yin für den Ton cis auf der A-Saite	106
8.9 Ausgabe der Funktion yin für Vierteltöne auf der E-Saite	107
8.10 Ausgabe der Funktion yin für Vierteltöne auf der E-Saite mit „plotthres- hold“=0,7	108
8.11 Ausgabe der Funktion yin für Vierteltöne auf der A-Saite	109

8.12 Ausgabe der Funktion <i>yin</i> für Vierteltöne auf der A-Saite mit Leistungsschwellwert=3,6 % der maximalen Leistung	110
8.13 Ausgabe der Funktion <i>yin</i> für Achteltöne auf der A-Saite	111
8.14 Ausgabe der Funktion <i>yin</i> für Achteltöne auf der A-Saite gespielt mit Fingerstyle	112
8.15 Ausgabe der Funktion <i>yin</i> für die A-Moll-Pentatonik aufgenommen mit der E-Tonabnehmerspule	113
8.16 Ausgabe der Funktion <i>yin</i> für die A-Moll-Pentatonik aufgenommen mit der A-Tonabnehmerspule	114
8.17 Ausgabe der Funktion <i>yin</i> für das Intervall A-dis aufgenommen mit der E-Tonabnehmerspule	115
8.18 Ausgabe der Funktion <i>yin</i> für das Intervall A-dis aufgenommen mit der A-Tonabnehmerspule	116
8.19 Ausgabe der Funktion <i>yin</i> für das Intervall a-g aufgenommen mit der E-Tonabnehmerspule	117
8.20 Ausgabe der Funktion <i>yin</i> für das Intervall a-g aufgenommen mit der A-Tonabnehmerspule	118
8.21 Ausgabe der Funktion <i>yin</i> für das Intervall H-fis/e aufgenommen mit der E-Tonabnehmerspule	119
8.22 Ausgabe der Funktion <i>yin</i> für das Intervall H-fis/e aufgenommen mit der A-Tonabnehmerspule	120
8.23 Ausgabe der Funktion <i>yin</i> für verschiedene Intervalle aufgenommen mit der E-Tonabnehmerspule	121
8.24 Ausgabe der Funktion <i>yin</i> für verschiedene Intervalle aufgenommen mit der A-Tonabnehmerspule	122
8.25 Polyphone Melodie mit gebrochenen Akkorden und Basstönen	123
8.26 Ausgabe der Funktion <i>yin</i> für die polyphone Melodie aufgenommen mit der E-Tonabnehmerspule	124
8.27 Ausgabe der Funktion <i>yin</i> für die polyphone Melodie aufgenommen mit der A-Tonabnehmerspule	125
8.28 Geöffnete Datei <i>EA2-solo.mid</i> im MIDI-Editor	127
8.29 Geöffnete Datei <i>EA2-solo.mid</i> im MIDI-Editor, Track 0 ausgeblendet . . .	128
8.30 Geöffnete Datei <i>EA2-solo.mid</i> im MIDI-Editor, letztes Event markiert . .	129
8.31 Geöffnete Datei <i>polyphoneMelodie.mid</i> im MIDI-Editor	130
8.32 Detaillierte Aufteilung der Programmlaufzeit der YIN-Analyse für 1.152 Datensamples	132

8.33 Detaillierte Aufteilung der Programmlaufzeit der YIN-Analyse für 571.747 Datensamples	133
8.34 Detaillierte Aufteilung der Programmlaufzeit der YIN-Analyse und MIDI-Konvertierung für 1.152 Datensamples	134
8.35 Detaillierte Aufteilung der Programmlaufzeit der YIN-Analyse und MIDI-Konvertierung für 571.747 Datensamples	135
10.1 Zusammengebauter Tonabnehmer des betreuenden Erstprüfers	147
A.1 Ablaufdiagramm der Funktion <i>yin</i>	161
A.2 Ablaufdiagramm der Funktion <i>yink</i>	162
A.3 Ablaufdiagramm der Funktion <i>yin_helper</i>	163
A.4 Ablaufdiagramm der Funktion <i>midiConverter</i>	164
A.5 Ablaufdiagramm der Funktion <i>writeVarLen</i>	165
A.6 Ablaufdiagramm der Funktion <i>writeBytes</i>	166
A.7 Ausgabe der Funktion <i>yin</i> für den Ton F auf der E-Saite	167
A.8 Ausgabe der Funktion <i>yin</i> für den Ton A auf der E-Saite	168
A.9 Ausgabe der Funktion <i>yin</i> für den Ton e auf der E-Saite	168
A.10 Ausgabe der Funktion <i>yin</i> für den Ton A auf der A-Saite	169
A.11 Ausgabe der Funktion <i>yin</i> für den Ton Ais auf der A-Saite	169
A.12 Ausgabe der Funktion <i>yin</i> für den Ton a auf der A-Saite	170
A.13 Ausgabe der Funktion <i>yin</i> für den Ton g' auf der A-Saite	170
A.14 Ausgabe der Funktion <i>yin</i> für den Ton E auf der E-Saite aufgenommen mit der A-Tonabnehmerspule	171
A.15 Ausgabe der Funktion <i>yin</i> für den Ton Ais auf der E-Saite aufgenommen mit der A-Tonabnehmerspule	171
A.16 Ausgabe der Funktion <i>yin</i> für den Ton A auf der A-Saite aufgenommen mit der E-Tonabnehmerspule	172
A.17 Ausgabe der Funktion <i>yin</i> für den Ton cis auf der A-Saite aufgenommen mit der E-Tonabnehmerspule	172

Tabellenverzeichnis

2.1	Die ersten drei Obertonfrequenzen der Grundtöne der Gitarre in Hz	9
2.2	Aufbau der Voice-Nachrichten „Note Off“, „Note On“ und „Program Change“	27
2.3	Beispiele für die variable Längenangabe der Delta-Zeit	31
2.4	Minimalbeispiel einer MIDI-Datei	32
4.1	Funktionale Anforderungen an den Tonabnehmer	55
4.2	Nicht-funktionale Anforderungen an den Tonabnehmer	55
4.3	Funktionale Anforderungen an das Instrumentenkabel	56
4.4	Nicht-funktionale Anforderungen an das Instrumentenkabel	56
4.5	Funktionale Anforderungen an die Vorverarbeitung	57
4.6	Funktionale Anforderungen an die MIDI-Konvertierung	58
4.7	Nicht-funktionale Anforderungen an die MIDI-Konvertierung	59
4.8	Nicht-funktionale Anforderungen an das gesamte System	59
5.1	Vergleich geeigneter E-Gitarren	63
5.2	Vergleich geeigneter Audiointerfaces	64
7.1	Kontaktbelegung des DIN-Steckers (von vorne betrachtet)	94
8.1	Beispielausgabe der Funktion <i>midiConverter</i>	123
8.2	Cell-Array mit Tonnummern, Lautstärken und Tonlängen als VLA	124
9.1	Erfüllungsgrad der funktionalen Anforderungen an den Tonabnehmer	138
9.2	Erfüllungsgrad der nicht-funktionalen Anforderungen an den Tonabnehmer	138
9.3	Erfüllungsgrad der funktionalen Anforderungen an das Instrumentenkabel	139
9.4	Erfüllungsgrad der nicht-funktionalen Anforderungen an das Instrumenten-	
	kabel	139
9.5	Erfüllungsgrad der funktionalen Anforderungen an die Vorverarbeitung	140

9.6	Erfüllungsgrad der funktionalen Anforderungen an die MIDI-Konvertierung	141
9.7	Erfüllungsgrad der nicht-funktionalen Anforderungen an die MIDI-Konvertierung	142
9.8	Nicht-funktionale Anforderungen an das gesamte System	143
A.1	MIDI-Nummern und dazugehörige Tonnamen und Frequenzen [56]	155

Abkürzungen

AlNiCo Legierung aus Aluminium, Nickel und Kobalt.

ASCII American Standard Code for Information Interchange.

ASIO Audio Stream Input/Output.

AWG American Wire Gauge.

CSV Comma-separated values.

DAW Digital Audio Workstation.

FFT Fast Fourier Transformation.

ggT Größter gemeinsamer Teiler.

LSB Least significant bit.

MIDI Musical Instrument Digital Interface.

MMA MIDI Manufacturers Association.

MSB Most significant bit.

PDA Pitch Detection Algorithm.

VLA Variable Längenangabe.

WAV Waveform Audio File Format.

Symbolverzeichnis

A Formelzeichen der Fläche.

A SI-Einheit der Stromstärke.

B Formelzeichen der magnetischen Flussdichte.

Baud, Bd Einheit der Symbolrate.

Bit Einheit der Datenmenge.

BPM, Beats Per Minute Anzahl der musikalischen Schläge pro Minute.

Byte, B Einheit der Datenmenge, 1 Byte = 8 Bit.

C Formelzeichen der Kapazität.

c Ausbreitungsgeschwindigkeit von Wellen.

Cent Maßeinheit für Tonhöhenintervalle.

dB Einheit des Pegelverhältnisses.

dBu Einheit des Spannungspegelverhältnisses mit der Bezugsgröße $0 \text{ dBu} = \sqrt{600 \Omega \cdot 1 \text{ mW}} \approx 775 \text{ mV}$.

dBV Einheit des Spannungspegelverhältnisses mit der Bezugsgröße $0 \text{ dBV} = 0 \text{ V}$.

f Formelzeichen der Frequenz.

f_0 Formelzeichen der Grundfrequenz (engl. fundamental frequency).

F SI-Einheit der elektrischen Kapazität.

ft, ' Angloamerikanische Einheit der Länge, $1 \text{ ft} = 0,3048 \text{ m}$.

°C Einheit der Temperatur.

° Einheit des Winkels.

H SI-Einheit der Induktivität.

Hz SI-Einheit der Frequenz.

I, \hat{I} , i Formelzeichen des elektrischen Stroms.

in," Angloamerikanische Einheit der Länge, 1 in = 0,0254 m.

L Formelzeichen der Induktivität.

l Formelzeichen der Länge.

λ Formelzeichen der Wellenlänge.

m SI-Einheit der Länge.

μ_0 Formelzeichen der magnetischen Feldkonstante.

N Formelzeichen der Anzahl.

Ω Einheit des elektrischen Widerstandes.

ω Formelzeichen der Kreisfrequenz.

Φ Formelzeichen des magnetischen Flusses.

Q Formelzeichen der Resonanzgüte.

R Formelzeichen des elektrischen Widerstandes.

ρ Formelzeichen des spezifischen elektrischen Widerstandes.

s SI-Einheit der Zeit.

t Formelzeichen der Zeit.

T₀ Formelzeichen der Grundperiode (engl. fundamental period).

U, \hat{U} , U, u Formelzeichen der elektrischen Spannung.

V Einheit der elektrischen Spannung.

Z, Z Formelzeichen der Impedanz.

1 Einleitung

1.1 Motivation

Algorithmen zur Tonhöhenerkennung im Kontext von Sprach- und Musikdetektion werden schon seit den 60er Jahren entwickelt. Vor allem in der Musikproduktion werden sie eingesetzt, um Musik auf Instrumenten aufzunehmen und automatisch in lesbare Noten zu übersetzen. Eines der Hauptprobleme auf diesem Gebiet sind polyphone Signale, die mehr als einen Ton zur selben Zeit beinhalten. Klassische Ansätze zur Analyse im Zeit- und Frequenzbereich können zwar monophone Signale präzise entschlüsseln, scheitern aber am zweiten Ton oder einer Geräuschkulisse. Für Gitarren gibt es deswegen spezielle Tonabnehmer, die für jede Gitarrensaite eine eigene Tonspur erzeugen, die dann mit monophonen Verfahren analysiert werden können.

1.2 Zielsetzung

Ziel dieser Arbeit ist es, einen hexaphonischen Tonabnehmer für eine E-Gitarre zu bauen und eine Verarbeitungskette von Tonerzeugung bis zur Tonhöhenerkennung zu entwickeln. Dieser Aufbau soll auch zukünftigen Arbeiten im Bereich Signalverarbeitung und maschinellen Lernen als Grundlage dienen. Die hexaphonisch getrennten Tonspuren sollen in dieser Arbeit in das branchenübliche MIDI-Format übersetzt werden. Dafür wird ein monopheron Algorithmus zur Tonhöhenerkennung eingesetzt und damit getestet, ob er so quasi-polyphon analysieren kann.

2 Grundlagen

In diesem Kapitel werden die benötigten Grundlagen zum Thema Gitarren und Tonabnehmer aufgeführt. Dafür wird zunächst der Aufbau und die Klangerzeugung von Gitarren erläutert. Es folgen eine kurze Erläuterung zur Notation von Gitarrennoten und spezieller Spieltechniken. Anschließend gibt es eine Einführung in die Klangbildung durch Obertöne gefolgt von der Technik der Tonabnehmer für E-Gitarren. Danach folgt der Aufbau von Instrumentenkabeln und welchen Einfluss sie auf den Gitarrenklang haben können. Anschließend werden die Ergebnisse eines Papers vorgestellt, dessen Autoren gemessen haben, aber wann eine Latenz zwischen Tonanschlag und Tonhören von Musikern als störend empfunden wird im Kontext von Echtzeit-Signalverarbeitung. Danach erfolgt eine Zusammenfassung der gängigen Ansätze zur Erkennung von Tonhöhen. Zuletzt wird der Aufbau und der Einsatz des MIDI-Formats erläutert.

2.1 Aufbau einer Gitarre

Gitarren bestehen im Wesentlichen aus drei Teilen: Korpus, Hals und Kopf, siehe Abbildung 2.1 [22]. Bei Gitarren mit elektrischer Klangerzeugung ist der Korpus massiv während er bei Gitarren mit akustischer Klangerzeugung hohl ist und ein Schalldoch besitzt [25]. Auf dem Korpus befindet sich der Steg, an dem die meist sechs Saiten befestigt sind. Sie verlaufen über das Schalldoch respektive die Tonabnehmer über das Griffbrett bis zum Kopf der Gitarre, wo sie mit Stimmwirbeln zum Stimmen fester oder loser gespannt werden [22]. Die sechs Saiten einer Gitarre heißen -beginnend bei der musikalisch tiefsten- E-, A-, d-, g-, h- und e'-Saite [22]. Auf dem Griffbrett der Gitarre befinden sich Bundstäbchen, die als Orientierung beim Spiel dienen, um die richtige Tonhöhe zu treffen. Eine Gitarre mit zum Beispiel 22 Bünden hat einen regulären Tonumfang von E - d'' (82,4 - 1174,6 Hz). Der erste Auflagepunkt der Saiten am oberen Ende des Halses wird 0. Bund genannt. Der Abstand zwischen dem 0. Bund und dem Gitarrensteg heißt Mensur und beschreibt die maximal frei schwingende Länge der Saite wenn sie gespielt

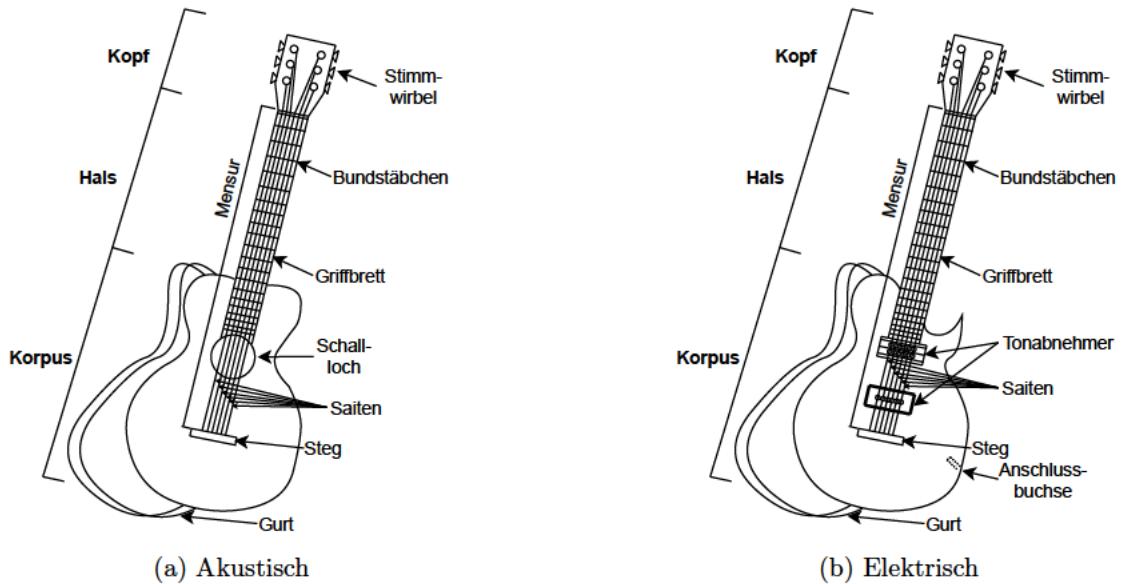


Abbildung 2.1: Wesentliche Bestandteile einer Gitarre

wird [22]. Bei E-Gitarren gibt es im Korpus eine Anschlussbuchse um sie mittels eines Instrumentenkabels an zum Beispiel einen Lautsprecher anzuschließen. Um eine Gitarre auch im Stehen spielen zu können, kann man am Korpus einen Gurt anbringen und diesen über die Schulter legen.

2.2 Klangerzeugung

Die Klangerzeugung bei Gitarren kann entweder akustisch, elektrisch oder durch eine Kombination beider Techniken erfolgen. Die Techniken werden im Folgenden beschrieben.

2.2.1 Akustisch

Bei akustischen Gitarren werden die gespannten Saiten angezupft oder angeschlagen und so in Schwingung versetzt. Die Saiten übertragen ihre Schwingung auf den Korpus, der als Resonanzkörper die Schwingung verstärkt und die ihn umgebende Luft in Schwingung versetzt. Letztendlich nimmt das menschliche Ohr diese Luftschnigungen als Klang wahr [22].



Abbildung 2.2: Gitarrennoten im Notensystem notiert

2.2.2 Elektrisch

Bei der elektrischen Klangerzeugung werden die Saitenschwingungen von metallischen beziehungsweise metall-umwickelten Saiten über Tonabnehmer in elektrische Spannung gewandelt und per Instrumentenkabel zu einem Lautsprecher geführt, über den man die gespielten Klänge hören kann. Die Töne werden vor dem Lautsprecher vorverstärkt oder mit Effekten bearbeitet, um die Klangfarbe und -art zu beeinflussen [57].

2.3 Notation von Gitarrennoten

Für Gitarrennoten gibt es zwei weit verbreitete Notationsformen, die im Folgenden erklärt werden.

2.3.1 Notensystem

Im klassischen Notensystem werden Gitarrennoten als kleine Ovale, sogenannte Notenköpfe, auf fünf Linien dargestellt, siehe Abbildung 2.2. Die vertikalen und horizontalen Linien am Notenkopf und sein Füllgrad geben die musikalische Länge eines Tones an. Das Symbol zu Beginn einer Notenzeile gibt an, welcher Referenzton auf welcher Linie zu finden ist [15]. In Abbildung 2.2 markiert der Notenschlüssel die zweitniedrigste Linie und gibt an, dass dort der Ton g notiert wird.

2.3.2 Tabulatur

Die Tabulatur wird verwendet, um das Spielen der Gitarre einfacher zu notieren. Ein Ton, notiert im Notensystem, kann auf verschiedenen Saiten gespielt werden. Das heißt beim Gitarrenspiel muss man sich entscheiden, wo man geschickt die Finger der linken

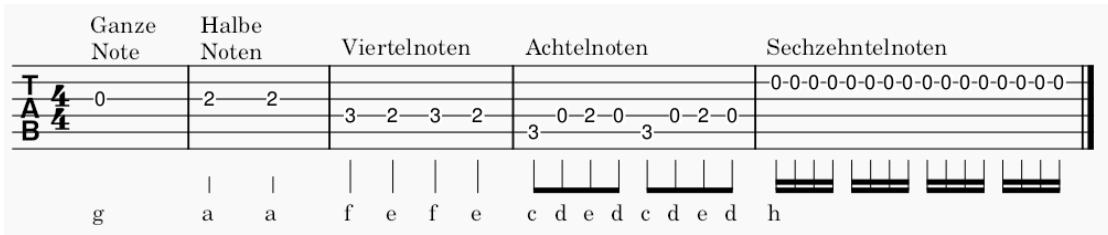


Abbildung 2.3: Gitarrennoten in einer Tabulatur notiert

Hand platziert, um so viele zu spielende Noten wie möglich abzudecken, ohne die linke Hand ständig bewegen zu müssen. Bei Gitarren kann man deswegen Noten auch in einer Tabulatur mit sechs Linien notieren, siehe Abbildung 2.3. Jede Linie steht für eine Saite, wobei die niedrigste Linie meistens der E-Saite entspricht [14]. Auf den Tabulatur-Linien werden Zahlen eingetragen, die dem zu spielenden Bund auf der Gitarre entsprechen. Damit wird festgelegt, wo welcher Ton gespielt werden soll. Die musikalische Tonlänge wird in Anlehnung an das Notensystem mit vertikalen und horizontalen Linien unterhalb der Tabulatur angedeutet. Sowohl das klassische Notensystem als auch Tabulaturen können als MusicXML-Dateien erzeugt werden, um sie zwischen gängigen Notationsprogrammen auszutauschen und zu editieren [17].

2.4 Spieltechniken

Beim Spiel der Gitarre gibt es diverse Spieltechniken, die einen Einfluss auf den Klang und den Verlauf von Tönen und Tonübergängen haben können. Diese Spieltechniken werden mit speziellen Symbolen innerhalb Gitarrennoten dargestellt, siehe Abbildung 2.4. Die bekanntesten Techniken werden im Folgenden erläutert.



Abbildung 2.4: Gitarrennoten mit eingetragenen Spieltechniken

Plektrum: Vor allem bei E-Gitarren wird häufig mit einem Plektrum gespielt [15]. Dabei handelt es sich um eine kleine, meist dreieckige Scheibe aus dünnem Plastik, die zwischen die Fingerknöchel des Daumen und Zeigefingers geklemmt wird. Mit dem Plektrum zwischen den Fingern werden die Saiten angeschlagen. Der Anschlag mit dem Plektrum erzeugt eine lokal starke Auslenkung und der Ton klingt hart [15].

Fingerstyle: Beim Fingerstyle werden die Saiten mit den Fingern der rechten Hand angezupft respektive angeschlagen [15]. Die Saiten werden dabei über eine größere Fläche als beim Plektrum ausgelenkt und die erzeugten Töne klingen weicher.

Bending: Beim Bending wird mit einem Finger der linken Hand die Saite auf Höhe eines der Bundstäbchens auf das Griffbrett gedrückt und mit der rechten Hand die Saite angezupft oder mit einem Plektrum angeschlagen. Anschließend schiebt der Finger der linken Hand die gedrückte Saite in Richtung einer der Nachbarsaiten und ändert damit die Saitenspannung [15]. Der Ton klingt höher, da durch das Bending die Ausbreitungs geschwindigkeit der Saite steigt.

Hammer On und Pull Off: Bei Hammer On und Pull Off werden zwei Töne nacheinander erzeugt, ohne die Saite ein zweites Mal anzuschlagen [15]. Für ein Pull Off werden die Finger der linken Hand auf zwei verschiedenen Bundstäbchen der gleichen Saite platziert und die Saite angeschlagen. Anschließend wird der Finger auf dem höheren Bundstäbchen kräftig nach oben gelöst, sodass jetzt der Ton des niedrigeren Bundstäbchen nachklingt. Beim Hammer On ist der Ablauf genau andersherum. Es wird ein Ton angeschlagen und auf derselben Saite auf einem höheren Bundstäbchen ein Finger kräftig aufgesetzt, sodass der höhere Ton mit dem gleichen Anschlag nachklingt.

Slide: Beim Slide wird ein Ton angeschlagen und der Finger der linken Hand anschließend zu einem höheren oder tieferen Bundstäbchen geschoben, sodass ein Anstieg oder Abfall der Tonhöhe zu hören ist [15].

Vibrato: Beim Vibrato wird wie beim Bending ein Ton angeschlagen und die niedergedrückte Saite in Richtung der Nachbars Seiten geschoben. Im Gegensatz zum Bending erfolgt das Schieben kontinuierlich hin und zurück, sodass ein Schwanken der Tonhöhe um den Ausgangston herum zu hören ist [15].

Wechselschlag: Wenn die Gitarre mit einem Plektrum gespielt wird, können die Saiten mit einem sogenannten Wechselschlag gespielt werden [15]. Dies wird vor allem bei hohem Tempo und schnellen Tonwechseln getan. Dabei wird das Plektrum erst von oben gegen die Saite geschlagen, überstreicht diese und das Plektrum endet nach dem Anschlag unterhalb der Saite. Beim nächsten Ton wird das Plektrum von unten gegen die Saite geschlagen, überstreicht diese und endet nach dem Anschlag wieder oberhalb der Saite. Diese Technik spart Zeit, da die Spielhand samt Plektrum nicht nach jedem Ton wieder in die Ausgangsposition gesetzt werden muss, um jeden Ton von oben respektive von unten anzuschlagen [15].

2.5 Obertöne

Die von einer Gitarrensaite erzeugten Töne bestehen nicht nur aus einem Ton sondern aus mehreren überlagerten Tönen, die vom menschlichen Ohr als einen zusammenhängenden Klang wahrgenommen werden [20]. Dies wird durch die Reflexion der Saitenbewegung erzeugt. Die Gitarrensaite ist an ihren beiden Enden fest an der Gitarre fixiert, sodass erzeugte Schwingungen an diesen beiden Punkten reflektiert werden und mit den einlaufenden Schwingungen so interferieren, dass sich stehende Wellen ausbilden. Die Wellenlängen bilden dabei ganzzahlige Bruchteile der zur Verfügung stehenden Saitenlänge. Für eine Mensur M hat der Grundton des erzeugten Klanges eine Wellenlänge von $\lambda_0 = 2 \cdot M$, wie in Abbildung 2.5 schematisch dargestellt. Der erste Oberton hat eine Wellenlänge von $\lambda_1 = 1 \cdot M$, der zweite Oberton eine Wellenlänge von $\lambda_2 = 1/2 \cdot M$ usw [11]. Die eingezeichneten Positionen des Hals- und Stegtonabnehmers entsprechen denen in der E-Gitarre aus Kapitel 5.2.1. Die Frequenzen der Grundtöne und der ersten drei Obertöne der Gitarren-Saiten sind in Tabelle 2.1 aufgeführt. Das ideale Frequenzspektrum eines Gitarrenklangs ist ein Linienspektrum, wobei die Amplitude mit steigender Frequenz der Obertöne sinkt, wie in Abbildung 2.6 schematisch dargestellt [26]. Ausschlaggebend für die wahrgenommene Tonhöhe ist die Frequenz des Grundtons. Werden beim Spiel die Finger auf die Saiten gesetzt, verkürzt sich die für die Schwingung verfügbare Saitenlänge, die Wellenlänge der Teiltöne sinkt und die Tonfrequenzen steigen. Dies geschieht auch, wenn die Saite beim Stimmen fester gespannt wird, da bei einer stärker gespannten Saite die Ausbreitungsgeschwindigkeit c der Schwingung steigt [19]:

$$f = \frac{c}{\lambda} \quad (2.1)$$

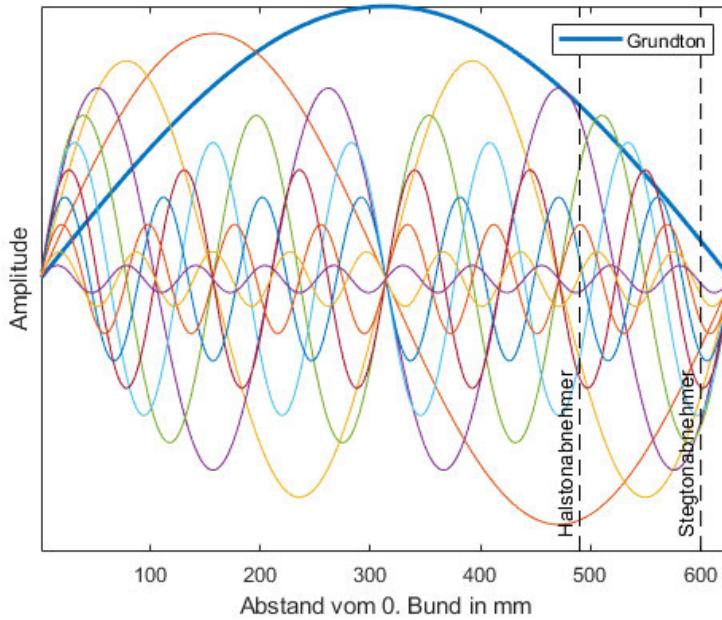


Abbildung 2.5: Idealer räumlicher Verlauf der Saitenschwingungen über eine Mensur von 628 mm

Je nach Position des Tonabnehmers gibt es Obertöne und deren ganzzahlige Vielfache, die gerade über dem Tonabnehmer einen Wellenknoten ausbilden und somit nicht erfasst werden können [25]. Bei Humbucker-Tonabnehmern, siehe Kapitel 2.6.2, können kurzwellige Obertöne über der einen Spule einen Wellenberg und über der anderen Spule ein Wellental ausbilden und löschen sich bei der Summation der Tonsignale aus [25].

Tabelle 2.1: Die ersten drei Obertonfrequenzen der Grundtöne der Gitarre in Hz nach [25]

	Grundton	1. Oberton	2. Oberton	3. Oberton
E	82,4	168,8	247,2	329,6
A	110	220	330	440
d	146,8	293,6	440,4	587,2
g	196	392	588	784
h	246,9	493,8	740,7	987,6
e'	329,6	659,2	988,8	1318,4

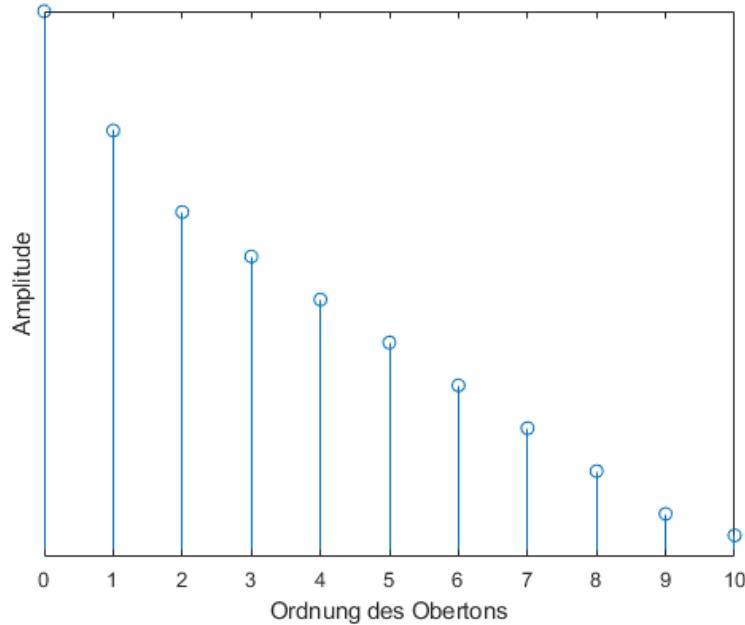


Abbildung 2.6: Ideales Linienspektrum für zehn Obertöne

2.6 Tonabnehmer

Tonabnehmer wandeln Schwingungen magnetischer Saiten in elektrische Spannung um. Die grundlegende Funktionsweise ist bei den meisten Tonabnehmern die gleiche, wobei sie sich in ihrer Bauform und ihrem Resonanzverhalten stark unterscheiden können. Im Folgenden wird die theoretische Funktionsweise erläutert, gefolgt von der Beschreibung der häufigsten Bauarten und dem Resonanzverhalten.

2.6.1 Funktionsweise

Ein Tonabnehmer besteht meistens aus einer Spule mit einem oder mehreren Permanentmagneten im Spulenkern. Diese Magneten erzeugen ein Magnetfeld mit konstanter Flussdichte \vec{B} . Integriert man \vec{B} über ein Flächenelement \vec{A} erhält man den magnetischen Fluss Φ durch dieses Flächenelement [21]:

$$\Phi = \int_A \vec{B} \cdot d\vec{A} \quad (2.2)$$

Wird eine der magnetischen Gitarrensaiten angeschlagen, bewegt sie sich im Magnetfeld der Permanentmagneten und beeinflusst die magnetische Flussdichte \vec{B} und gemäß Gleichung 2.2 auch den magnetischen Fluss Φ . Dabei hat nur die senkrechte Komponente der Bewegung einen Einfluss. Nach dem Faradayschen Induktionsgesetz führt eine Änderung des magnetischen Fluxes Φ durch eine von einem elektrischen Leiter umschlossenen Fläche zu einer induzierten Spannung U_{ind} in diesem Leiter [21]. Die Richtung der induzierten Spannung ist dabei gemäß der Lenz'schen Regel der Änderungsrichtung des magnetischen Fluxes entgegengesetzt, verdeutlicht durch ein Minuszeichen [21]:

$$U_{ind} = -\frac{d}{dt}\Phi = -\frac{d}{dt} \int_A \vec{B} \cdot d\vec{A} \quad (2.3)$$

Im Kontext von Gitarrensaiten, die sich zum Tonabnehmer hin und wieder zurück bewegen, wird eine Wechselspannung induziert und die anfängliche Richtung der Spannung ist unerheblich. Da die Spule des Tonabnehmers aufgrund der N Wicklungen aus N von einem Leiter umschlossenen Flächen besteht, ergibt sich eine N -fache Ausgangsspannung [21]:

$$U_{ind,Spule} = -N \cdot \frac{d}{dt}\Phi \quad (2.4)$$

2.6.2 Bauformen

Die zwei gängigsten Bauformen für Magnettonabnehmer sind Single Coil und Humbucker. Außerdem gibt es speziell für MIDI-Anwendungen polyphone Tonabnehmer. Alternativ zu Magnettonabnehmern gibt es Piezo-Tonabnehmer. Die Bauformen werden im Folgenden näher beschrieben.

Single Coil

Single Coil-Tonabnehmer besitzen die einfachste Bauform. Sechs Stabmagneten gleicher Polarisierung sind auf einer Bodenplatte montiert und bilden den Spulenkern, um den der Spulendraht gewickelt wird. Alternativ befindet sich in der Bodenplatte ein Balkenmagnet, auf dem sechs Eisenstifte oder -schrauben stehen und durch den Balkenmagnet polarisiert werden. Die Spule wird üblicherweise durch ein Gehäuse vor Beschädigungen geschützt. Besteht der Tonabnehmer aus Eisenschrauben, kann man diese tiefer oder weiter aus dem Gehäuse schrauben und dadurch die Empfindlichkeit des Tonabnehmers

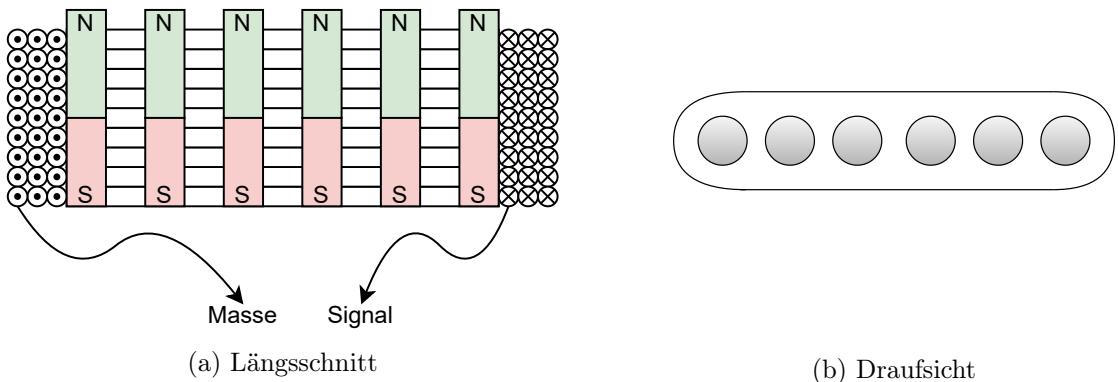


Abbildung 2.7: Einfacher Aufbau eines Single Coil-Tonabnehmers

für jede Saite individuell anpassen. Üblicherweise wird das innen liegende Drahtende der Spule als „Hot“ bezeichnet und das äußere als „Cold“ . Der äußere Anschluss wird meistens mit dem Massepotential verbunden. Die innen liegenden Wicklungen werden so durch die äußeren Wicklungen vor Störungen durch elektrische Felder abgeschirmt [43]. Die Erzeugung des elektrischen Signals erfolgt nach dem Induktionsgesetz beschrieben in Kapitel 2.6.1. Single Coil-Tonabnehmer haben den Nachteil, dass sie nicht nur auf die Änderung des statischen Magnetfeldes reagieren, sondern auch auf Magnetfelder ferner Quellen von zum Beispiel Netzteilen, Stromkabeln oder Leuchtstoffröhren. Diese Magnetfelder induzieren in der Spule eine störende Brummspannung [57]. Der schematische Aufbau eines Single Coil-Tonabnehmers ist in Abbildung 2.7 zu sehen.

Humbucker

Humbucker-Tonabnehmer können durch ihren Aufbau die störende Brummspannung Magnetfelder ferner Quellen unterdrücken. Sie bestehen aus zwei Einzelspulen, die gleichpolig in Reihe geschaltet sind. Ihre Magnet- beziehungsweise Eisenstifte sind aber gegensätzlich polarisiert. Magnetfelder ferner Quellen induzieren in beiden Spulen die gleiche Brummspannung, die sich durch die gleichpolige Reihenschaltung zu 0 aufhebt, siehe Abbildung 2.9a. Wird aber eine Saite der E-Gitarre angeschlagen, beeinflusst sie die Magnetfelder der gegenüberliegenden Magneten und die induzierte Spannung verdoppelt sich durch die gleichpolige Reihenschaltung, siehe Abbildung 2.9b [29]. Der schematische Aufbau des Humbucker-Tonabnehmers ist in Abbildung 2.8 zu sehen.

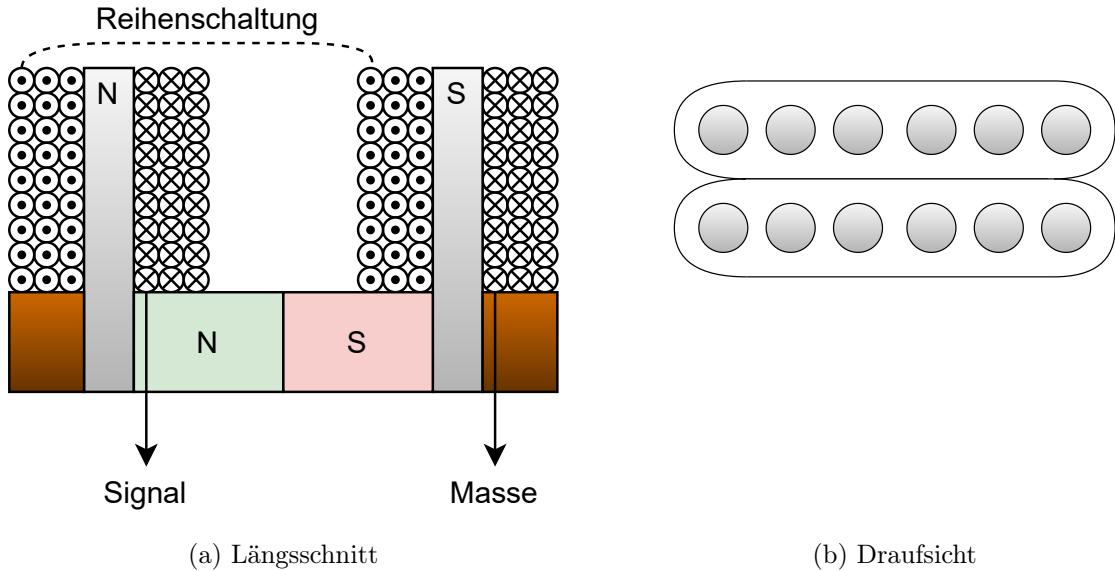


Abbildung 2.8: Aufbau eines Humbucker-Tonabnehmers

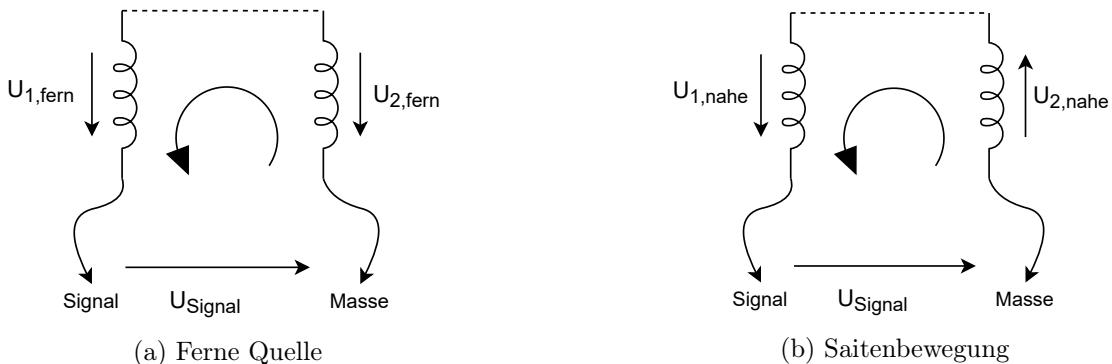


Abbildung 2.9: Induzierte Spannungen am Humbucker-Tonabnehmer nach [29]

Polyphon

Polyphone Tonabnehmer besitzen je Saite eine eigene Spule. Im Fall von E-Gitarren meist sechs Spulen. Polyphone Tonabnehmer für Gitarren werden deswegen weit verbreitet hexaphonisch genannt. Durch die unabhängigen Spulen lassen sich die Signale der Saiten einzeln verarbeiten [26]. Hexaphonische Tonabnehmer werden vor allen in Gitarrensynthesizern und MIDI-Gitarren eingesetzt. Im Fall von MIDI-Gitarren können die Tonsignale der Saiten in das MIDI-Format konvertiert werden, wobei hierfür noch ein zusätzlicher Konverter hinter dem hexaphonischen Tonabnehmers benötigt wird [43].

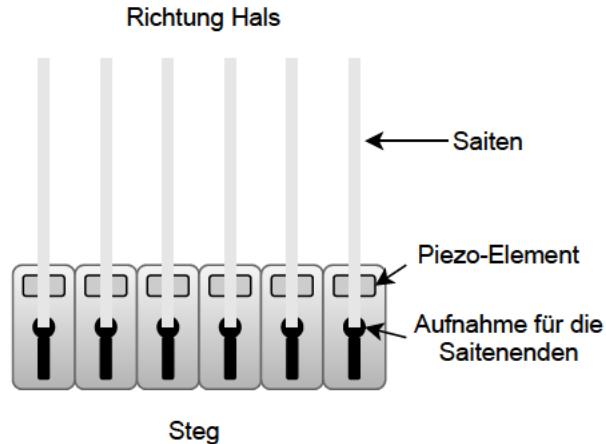


Abbildung 2.10: Aufbau eines Piezo-Stegtonabnehmers

Piezo

Piezo-Tonabnehmer verwenden keine Spulen zur Spannungserzeugung sondern kleine Plättchen aus piezoelektrischen Material, die in speziellen Stegonabnehmern fest eingebaut sind, siehe Abbildung 2.10 [25]. Die Saiten liegen direkt auf den Plättchen und übertragen beim Spiel Druck auf sie, sodass durch die Materialverformung Spannung erzeugt wird. Häufig werden Blei-Zirkonat-Titanat oder Polyvinylidenfluorid (PVDF) als Material verwendet [25]. Piezo-Tonabnehmer haben den Vorteil, dass sie auch bei Akustikgitarren ohne magnetische Saiten eingesetzt werden können. Eingebaut in E-Gitarren erzeugen sie einen Klang, der Akustikgitarren nahe kommt. Piezo-Tonabnehmer gibt es auch in hexaphonischer Form mit sechs getrennten Steg-Tonabnehmerteilen [6]. Alternativ zu Stegonabnehmern gibt es auch Piezo-Tonabnehmer, die auf den Korpus geklebt werden können und die Korpussschwingung aufnehmen. Sie haben den Nachteil, dass auch Berührungen beim Hantieren mit der Gitarre als Geräusch übertragen werden [25].

2.6.3 Resonanz

Die Induktivität L_1 , der Drahtwiderstand R_1 und die Wicklungskapazität C_1 eines Magnettonabnehmers bilden zusammen einen Reihenschwingkreis, dargestellt im Ersatzschaltbild in Abbildung 2.11. Das Verhältnis zwischen der Ausgangs- und der induzierten Spannung lässt sich mit einem Tiefpass 2. Ordnung und der Übertragungsfunktion $H(\omega)$

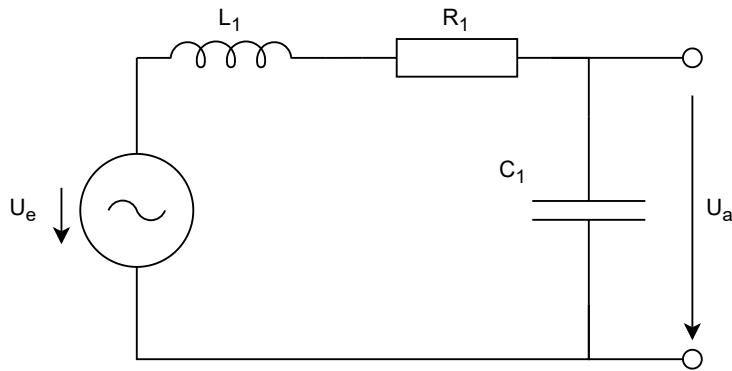


Abbildung 2.11: Ersatzschaltbild eines Tonabnehmers

beschreiben [57]:

$$\begin{aligned}
 H(\omega) &= \frac{\underline{U}_a}{\underline{U}_e} = \frac{\underline{Z}_{C1}}{\underline{R}_1 + \underline{Z}_{L1} + \underline{Z}_{C1}} = \frac{\frac{1}{j\omega C_1}}{\underline{R}_1 + j\omega L_1 + \frac{1}{j\omega C_1}} \\
 &= \frac{1}{1 + j\omega R_1 C_1 + \omega^2 L_1 C_1}
 \end{aligned} \tag{2.5}$$

Der Schwingkreis befindet sich in Resonanz, wenn sich die Blindwiderstände von L_1 und C_1 kompensieren und die Gesamtimpedanz rein reell ist [21]:

$$f_r = \frac{1}{2\pi \cdot \sqrt{L_1 C_1}} \tag{2.6}$$

Die niedrigen Frequenzen ab 0 Hz bis kurz vor f_r werden mit der Grundamplitude des Tonabnehmers übertragen, Frequenzen im Bereich um f_r werden verstärkt übertragen und die Frequenzen über f_r werden stark gedämpft [26, 57]. Die Höhe der Resonanzüberhöhung wird mit der Resonanzgüte Q beschrieben. Je höher die Güte, desto höher die Resonanzüberhöhung [57]. Q wird über das Verhältnis von Scheinwiderstand zu Verlustwiderstand im Resonanzfall berechnet [21]:

$$Q = \frac{\underline{Z}_{L1}(f_r)}{\underline{R}_1} = \frac{2\pi f_r \cdot L_1}{R_1} = \frac{1}{R_1} \cdot \frac{L_1}{\sqrt{L_1 C_1}} = \frac{1}{R_1} \cdot \sqrt{\frac{L_1}{C_1}} \tag{2.7}$$

L_1 und C_1 haben gemäß Gleichung 2.6 alleinigen Einfluss auf die Lage der Resonanzfrequenz, während R_1 gemäß Gleichung 2.7 einen großen Einfluss auf die Resonanzgüte hat. In Abbildung 2.12 ist zu erkennen, dass Q mit kleiner werdenden Werten für R_1 steigt.

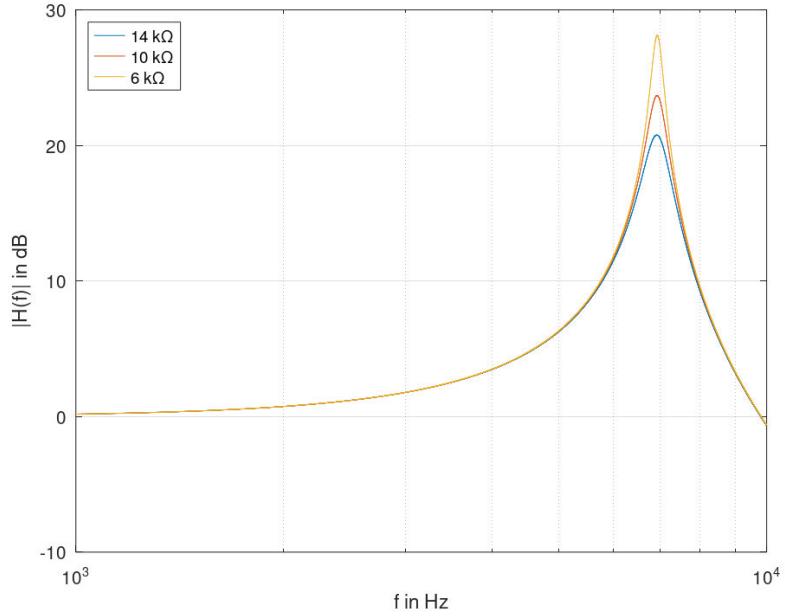


Abbildung 2.12: Variation von R_1 bei $C_1 = 150 \text{ pF}$ und $L_1 = 3.5 \text{ H}$

In den Abbildungen 2.13 und 2.14 auf den Seiten 17 und 18 ist zu erkennen, dass mit steigenden Werten für C_1 und L_1 f_r sinkt. Dabei lassen steigende Werte für C_1 Q sinken und steigende Werte für L_1 Q ansteigen. In der Realität ist ein Tonabnehmer zusätzlich mit einer externen Schaltung bestehend aus einem Instrumentenkabel und einem Verstärkereingangswiderstand belastet. Das führt zum Ersatzschaltbild in Abbildung 2.15 auf Seite 18. Der Längswiderstand im Instrumentenkabel aus Kapitel 7.2.1 beträgt $0,3 \Omega$ und kann als so gering betrachtet werden, dass sich das Ersatzschaltbild zu Abbildung 2.16 auf Seite 19 vereinfachen lässt, wenn man die Wicklungskapazität und die Kabelkapazität zur Kapazität C zusammenfasst. Das führt zur neuen Übertragungsfunktion $H'(\omega)$ und Q' [57]. Der Einfluss von R_3 ist in Abbildung 2.17 auf Seite 19 deutlich zu erkennen. Der Innenwiderstand R_1 und der Eingangswiderstand R_3 bilden einen Spannungsteiler für die im Tonabnehmer erzeugte Spannung U_e . Je kleiner R_3 ist, desto weniger Spannung

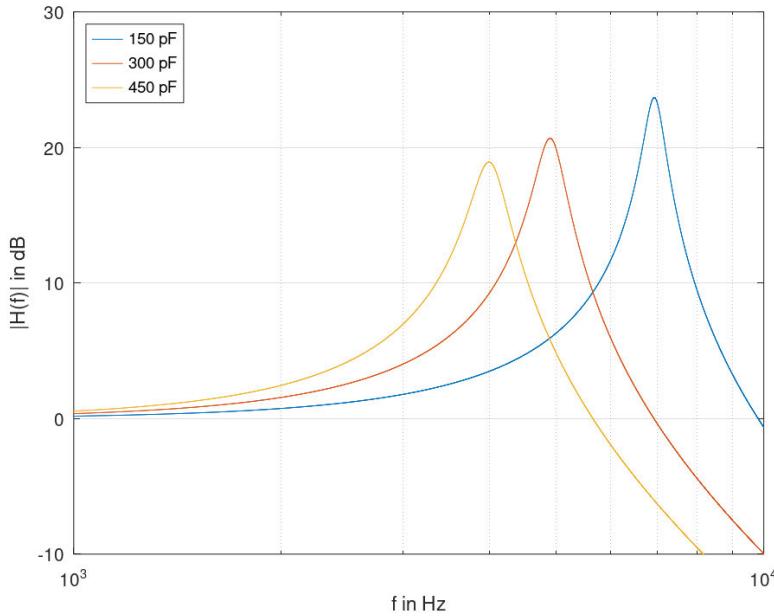


Abbildung 2.13: Variation von C_1 bei $R_1 = 10 \text{ k}\Omega$ und $L_1 = 3.5 \text{ H}$

fällt an ihm ab und die Resonanzgüte Q sinkt.

$$\begin{aligned}
 H'(\omega) &= \frac{U_a}{U_e} = \frac{\underline{Z}_C || R_3}{R_1 + \underline{Z}_{L1} + \underline{Z}_C || R_3} = \frac{\frac{R_3}{1 + j\omega R_3 C}}{R_1 + j\omega L_1 + \frac{R_3}{1 + j\omega R_3 C}} \\
 &= \frac{R_3}{R_1 + R_3 + j\omega(R_1 R_3 C + L_1) - \omega^2(R_3 L_1 C)} \\
 &= \frac{1}{1 + R_1/R_3 + j\omega(R_1 C + L_1/R_3) - \omega^2(L_1 C)}
 \end{aligned} \tag{2.8}$$

$$Q' = \frac{\sqrt{L_1 C (1 + R_1/R_3)}}{R_1 C + L_1/R_3} \tag{2.9}$$

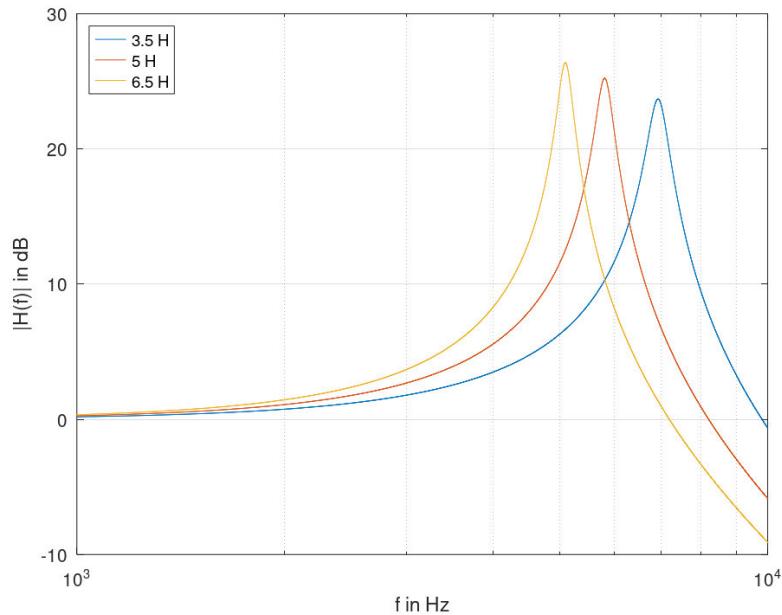
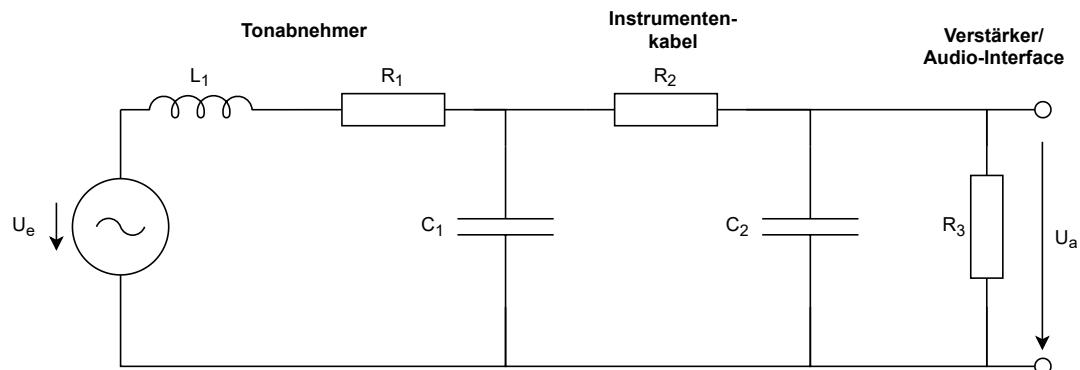

 Abbildung 2.14: Variation von L_1 bei $R_1 = 10 \text{ k}\Omega$ und $C_1 = 150 \text{ pF}$


Abbildung 2.15: Ersatzschaltbild eines Tonabnehmers mit externer Beschaltung

2.7 Passive Schaltung in einer E-Gitarre

Die meisten E-Gitarren verfügen über einen Lautstärke- und einen Klangregler, die direkt in der Gitarre verbaut sind [25]. Über den Lautstärkeregler wird die Ausgangslautstärke eingestellt und über den Klangregler wird die Amplitude im höheren Frequenzbereich beeinflusst [25]. Außerdem kann bei mehreren verbauten Tonabnehmern eingestellt wer-

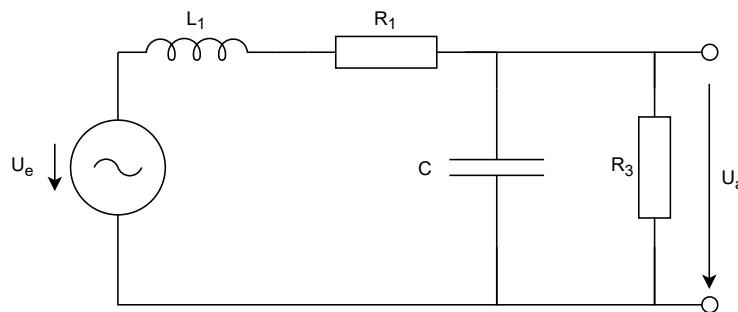


Abbildung 2.16: Ersatzschaltbild eines Tonabnehmers mit vereinfachter externer Beleuchtung

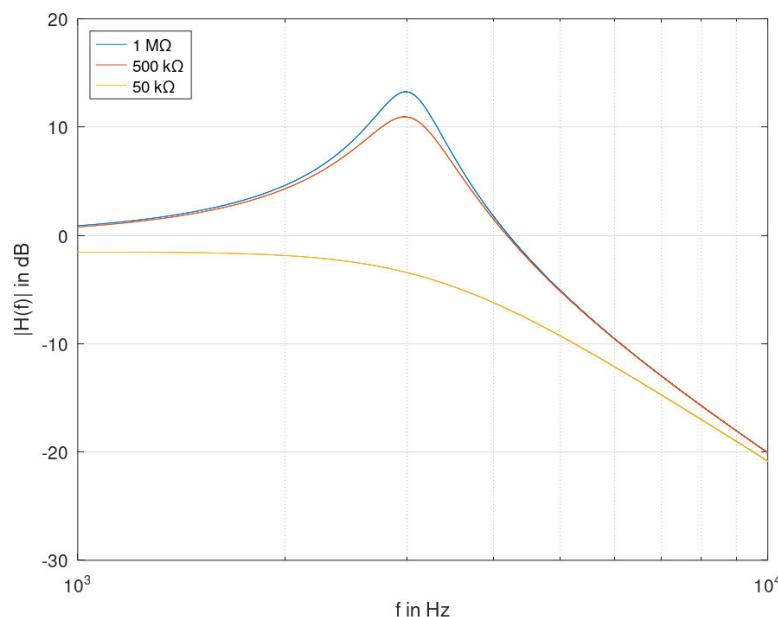


Abbildung 2.17: Variation von R_3 bei $R_1 = 10 \text{ k}\Omega$, $L_1 = 3.5 \text{ H}$ und $C = 800 \text{ pF}$

den, ob zum Beispiel einige parallel oder seriell geschaltet werden oder nur einer genutzt wird für die Tonausgabe. Die Schaltung der später verwendeten Gitarre aus Kapitel 5.2.1 ist in Abbildung 2.18 zu sehen. Die zwei Tonabnehmer können über den Kippschalter entweder alleine oder parallel genutzt werden. Darauf folgen der Klangregler bestehend aus einem Potentiometer und einem Kondensator und ein Potentiometer als Lautstärkeregler. Zuletzt ist die abgehende Klinkenbuchse angebracht. Der Lautstärkeregler hat

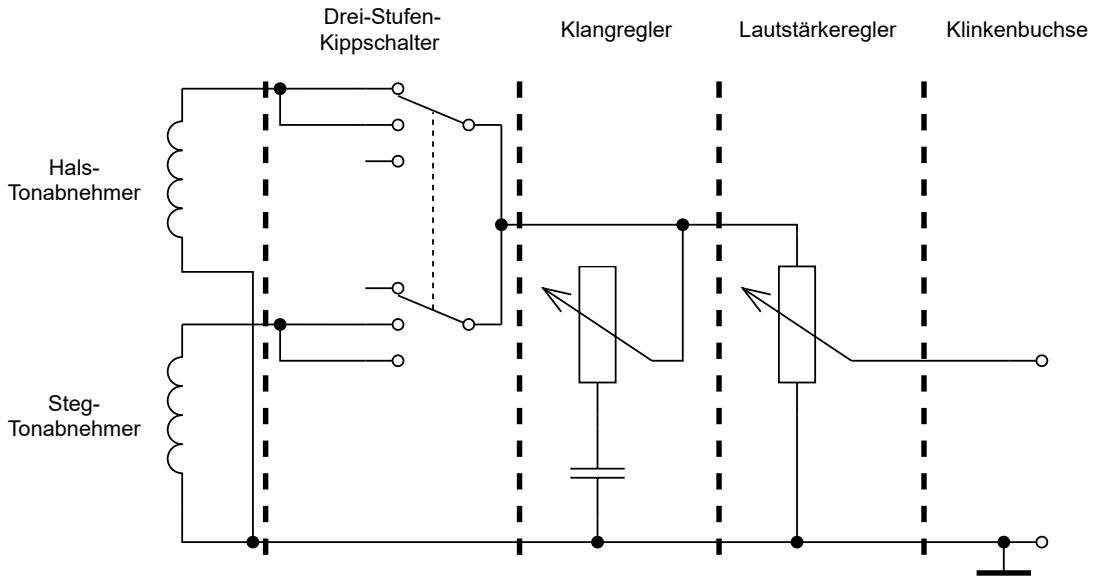


Abbildung 2.18: Passive Schaltung der E-Gitarre aus Kapitel 5.2.1

einen ähnlichen Einfluss wie R_3 in Abbildung 2.17: Wird der Schleifer des Potentiometers Richtung Masse gedreht, sinkt die Resonanzgüte [25].

2.8 Instrumentenkabel und Anschlusskontakte

Die Instrumentenkabel zur Verbindung von E-Gitarren mit Verstärkern oder Lautsprechern sind üblicherweise koaxial aufgebaut und übertragen das Tonsignal unsymmetrisch, siehe Kapitel 2.9 zur Erläuterung der Übertragungsarten [57]. Wie im Kapitel 2.6.3 bereits erwähnt, hat das Instrumentenkabel aufgrund seiner Längskapazität einen Einfluss auf die Resonanzfrequenz des Tonabnehmers. Da Instrumentenkabel selten unbewegt auf dem Boden liegen sondern regelmäßig aufgerollt, gebogen und sogar betreten werden, ist es wichtig, dass sie einen robusten Mantel und eine stabile Abschirmung haben. In hochwertigen Instrumentenkabeln besteht die Abschirmung aus geflochtenen Kupferlitzen, die beim Biegen und Bewegen ihre Form beibehalten. In qualitativ schlechteren Kabeln besteht die Abschirmung aus glatt umlaufenden Einzellitzen, die bei Bewegung verrutschen oder brechen und dadurch die Fläche der Abschirmung verändern und verschlechtern [26]. Instrumentenkabel sind beidseitig mit Klinkensteckern ausgestattet. Diese sind für gewöhnlich zweipolig für das Tonsignal und die Masseverbindung aufgebaut. Der In-

nenleiter und die Abschirmung des Instrumentenkabels werden an die innen liegenden Kontakte des Klinkensteckers gelötet und das Kabel mit einer Zugentlastung im Steckergehäuse fixiert. Am häufigsten werden Klinkenstecker mit einem Durchmesser von $1/4$ in (6,35 mm) eingesetzt, seltener mit einem Durchmesser von 3,5 mm [26]. Die Kontaktflächen der Stecker und Buchsen können mit der Zeit oxidieren und nicht mehr richtig leiten. Entweder man feilt die oxidierten Flächen ab oder investiert in teurere vergoldete Stecker und Buchsen, die nicht oxidieren können [26]. Einen Einfluss auf den Klang hat das Kontaktmaterial nicht.

2.9 Spannungspegel in der Tontechnik

In der Tontechnik werden vier verschiedene Spannungspegel beim Anschluss von Instrumenten, Mikrofonen, Mischpulten und Lautsprechern verwendet. Dabei ist es wichtig, dass Signale mit einem bestimmten Pegel an einen Eingangskontakt angeschlossen werden, der für diesen Pegel ausgelegt ist. Sonst kann das Signal zu leise oder übersteuert klingen oder weiterführende Elektronik zerstört werden [40]. Außerdem wird unterschieden, ob die Pegel symmetrisch oder unsymmetrisch übertragen werden. Bei der symmetrischen Übertragung gibt es zwei Signalleiter, die das Tonsignal mit 180° -Phasenlage zueinander übertragen. Auf der Eingangsseite wird die Differenz aus beiden Signalen gebildet, sodass Störungen auf dem Übertragungsweg, die auf beide Leiter mit der gleichen Amplitude einwirken, kompensiert werden [49]. Bei der unsymmetrischen Übertragung gibt es lediglich einen Signalleiter und eine Masseverbindung als Rückleiter, sodass Störungen auf dem Signalweg nicht herausgerechnet werden können [49]. Die vier Spannungspegel werden im Folgenden beschrieben.

2.9.1 Line-Pegel

Der Line-Pegel ist der Referenzpegel in der Tontechnik und beträgt für Heimanwendungen -10 dBV (316 mV) und im Studiobereich +4 dBu (1,23 V) [49]. Er wird bei der Verbindung von nicht-instrumentalen Geräten wie Verstärkern, Mischpulten oder Audio-Interfaces benutzt. Außerdem können einige Instrumente wie Synthesizer Line-Pegel ausgeben [40]. Die Eingangsimpedanz von Line-Eingängen beträgt typischerweise $10\text{ k}\Omega$, sie kann aber auch zwischen $600\text{ }\Omega$ und $100\text{ k}\Omega$ liegen [53]. Line-Pegel können je nach Ausgabegerät symmetrisch als auch unsymmetrisch übertragen werden [53].

2.9.2 Mikrofon-Pegel

Der Mikrofon-Pegel wird von Mikrofonen erzeugt. Der empfohlene Wert für die Ausgangsspannung beträgt laut DIN EN IEC 61938:2018-10 zwischen 0,2 mV und 0,2 V bei einer Ausgangsimpedanz von 200Ω für elektrodynamische Mikrofone [9]. Die Eingangsimpedanz von Mikrofon-Eingängen soll mindestens $1 k\Omega$ betragen [9]. Von Eingangsimpedanzen im $M\Omega$ -Bereich wird abgesehen, da Widerstände störendes Rauschen erzeugen können, wenn sie mit Strom durchflossen werden. Das Rauschen wird umso lauter, je größer der Widerstand ist [40]. Mikrofon-Pegel werden bei professionellen Geräten üblicherweise symmetrisch übertragen [53]. Da der Mikrofon-Pegel deutlich geringer als der Line-Pegel ist, müssen Mikrofonsignale durch einen Mikrofonverstärker oder eingangsseitig der Audiointerfaces verstärkt werden. Einige Mikrofone benötigen eine Spannungsversorgung durch das Mikrofonkabel, die sogenannte Phantom-Speisung. Sie beträgt üblicherweise 48 V. Die Phantomspeisung wird von den Audiointerfaces an den Eingangskontakten bereitgestellt und kann meistens manuell ein- oder ausgeschaltet werden [49]. Schließt man ein Mikrofon an einen Line-Eingang an, wird man das Signal kaum hören, da der Spannungspegel deutlich niedriger als erwartet ist. Schließt man umgekehrt eine Line-Quelle an einen Mikrofon-Eingang an, wird das Signal sehr laut und teilweise übersteuert klingen, da der Spannungspegel deutlich höher als erwartet ist.

2.9.3 Instrumenten-Pegel

Der Instrumenten-Pegel wird von Instrumenten wie E-Gitarren und E-Bässen erzeugt. Die Ausgangsspannung ist etwas höher als beim Mikrofon-Pegel, wobei die Ausgangsimpedanz von Tonabnehmern typischerweise im $k\Omega$ -Bereich liegt [25]. Dementsprechend liegen die Eingangsimpedanzen für Instrumenten-Eingänge im $M\Omega$ -Bereich, da sonst beim Spannungsteiler aus Ausgangsimpedanz im $k\Omega$ -Bereich und Eingangsimpedanz im $k\Omega$ -Bereich deutlich weniger Spannung an der Eingangsimpedanz abfällt. Der Instrumenten-Pegel wird, wie in Kapitel 2.8 beschrieben, unsymmetrisch übertragen. Schließt man eine E-Gitarre an einen Line- oder Mikrofon-Eingang an, ist sie deutlich leiser zu hören und der Resonanzverlauf wird deutlich flacher, siehe Abbildung 2.17. Ist die E-Gitarre am Instrumenten-Eingang angeschlossen, muss das Signal trotzdem wie beim Mikrofon auf den Line-Pegel verstärkt werden. Das kann eingangsseitig im Audiointerface erfolgen.

2.9.4 Lautsprecher-Pegel

Lautsprecher haben üblicherweise eine Eingangsimpedanz von 4Ω oder 8Ω [53]. Der Lautsprecher-Pegel liegt im zweistelligen V-Bereich. Lautsprecher verfügen entweder über eine aktive Eingangsverstärkung oder werden über einen separaten Lautsprecherverstärker angeschlossen, der eingangsseitig ein Signal mit Line-Pegel erhält.

2.10 Hörbare Latenz beim Live-Spiel

In den meisten Fällen schließt man E-Gitarren mindestens an einen Lautsprecher an, um sie beim Spiel besser hören zu können. Dabei wird es von den Musikern als sehr störend empfunden, wenn der Ton aus dem Lautsprecher deutlich später gehört wird, als die Bewegung des Tonanschlags erfolgt. In [27] wurde sich mit der Frage beschäftigt, ab welcher Zeitspanne Berufsmusiker eine Latenz wahrnehmen können. Es wurde ermittelt, dass beim Spiel von E-Gitarren eine Latenz ab 16 ms zwischen Tonanschlag und dem gehörten Ton eines 4 - 6 ft (1,2 - 1,8 m) entfernten Lautsprechers als hörbare Verzögerung wahrgenommen wird [27].

2.11 Algorithmen zur Tonhöhenerkennung

Algorithmen zur Tonhöhenerkennung (engl. Pitch Detection Algorithm (PDA)) werden in der Akustik und Signalverarbeitung eingesetzt, um die wahrgenommene Tonhöhe respektive -frequenz (engl. fundamental frequency) f_0 von Tönen und Tongemischen zu bestimmen [16]. Dazu wird oft die Grundperiode T_0 des Tons gesucht, deren Inverse f_0 entspricht. Außerdem werden PDA zur Spracherkennung und -transkription eingesetzt. Es gibt verschiedene Ansätze zur Bestimmung von f_0 , die Merkmale im Zeit- oder Frequenzbereich auswerten oder maschinelles Lernen und neuronale Netze einsetzen [16]. Bei PDA wird unterschieden, ob sie monophone oder auch polyphone Signale analysieren können. Polyphon bedeutet in diesem Fall, dass das Tonsignal zu einem Zeitpunkt aus mindestens zwei unterschiedlichen Tonhöhen und deren Obertönen gleichzeitig besteht, wie es bei der Aufnahme von Gitarren- oder Klavierakkorden der Fall ist. Viele PDA arbeiten nur monophon, da ihre zugrunde liegenden Ansätze von der Existenz von nur einer Tonhöhe zur Zeit ausgehen. Die Tonhöhenerkennung mittels PDA ist der erste Schritt zur automatischen Musiktranskription in das MIDI-Format oder direkt in lesbare

Musiknoten. Im Kontext von Live-Analyse es wichtig, dass die PDA eine so geringe Analyselatenz wie möglich aufweisen, wie in Kapitel 2.10 bereits erläutert. Die meisten PDA, die Merkmale im Zeit- oder Frequenzbereich analysieren, haben immer eine Grundlatenz, da mindestens zwei Perioden der tiefsten erwarteten Töne an Datenpunkten des Tonsignals aufgenommen werden müssen, um wichtige Merkmale wie Nulldurchgänge sicher zu erfassen.

2.12 MIDI-Format

Das Musical Instrument Digital Interface (MIDI) beschreibt ein Kommunikationsprotokoll zwischen Musikinstrumenten und Musiksteuergeräten mittels digitaler serieller Übertragung [47]. MIDI 1.0 wurde 1983 von mehreren Herstellern der elektronischen Musikinstrumentenbranche als branchenweiter Standard entwickelt und wird aktuell von der MIDI Manufacturers Association (MMA) gepflegt. MIDI 1.0 enthält Spezifikationen zum Datenformat und zu den eingesetzten Hardwareschnittstellen, die im Folgenden erläutert werden. Aus dem Datenformat für die Kommunikation zwischen MIDI-Geräten hat sich das MIDI-Dateiformat zur Speicherung von MIDI-Daten abgeleitet [48].

2.12.1 Hardwareschnittstelle

Die MIDI-Hardwareschnittstelle arbeitet unidirektional mit 31,25 ($\pm 1\%$) kBd, wobei ein übertragenes Symbol aus einem Startbit, acht Datenbit und einem Stopbit besteht. Das Startbit ist logisch 0 (Strom ein) und das Stopbit ist logisch 1 (Strom aus), die Bits werden mit dem Least significant bit (LSB) zuerst übertragen. Die Referenzspannung beträgt 5 V und der Strom 5 mA [47].

Es werden fünf-polige DIN-Stecker und -Buchsen für den Anschluss sowie Optokoppler respektive Transistoren oder IC als Schnittstellentreiber verwendet, siehe Abbildung 2.19 [47]. Alle verbauten Widerstände haben eine Toleranz von 5 % [47]. An den MIDI-Geräten werden DIN-Buchsen zur Frontplattenmontage verwendet, MIDI-Kabel haben an beiden Enden DIN-Stecker. Insgesamt gibt es drei Arten von MIDI-Anschlüssen: MIDI IN, MIDI THRU und MIDI OUT [47]. MIDI THRU wird zur Reihenschaltung mehrerer MIDI-Empfangsgeräte verwendet, wobei das empfangene Signal vom MIDI IN-Eingang direkt zum MIDI THRU-Ausgang parallel übertragen wird. Bei allen Anschlüssen sind die Pins 1 und 3 nicht angeschlossen, bei MIDI IN ist zusätzlich Pin 2 nicht angeschlossen [47].

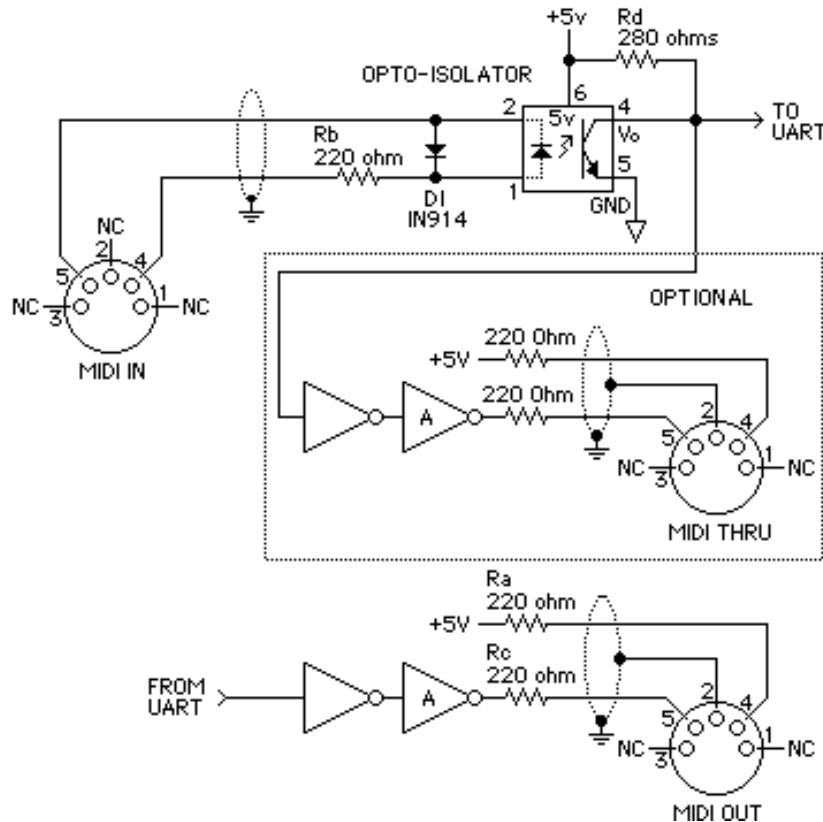


Abbildung 2.19: MIDI Standard Hardware [47]

Bei MIDI OUT und THRU wird Pin 2 mit dem Schirm des MIDI-Kabels verbunden, Pin 4 übt die 5 V-Referenzspannung und Pin 5 die Signalpegel. MIDI-Kabel sind Kabel mit geschirmten verdrillten Doppeladern, deren Schirmung nur einseitig bei den MIDI-Ausgängen aufgelegt ist. Das Schirmpotential ist getrennt von sonstigen Gehäuse- oder Schaltungsmassepotentialen [47].

2.12.2 Datenformat

Im Folgenden werden der Aufbau und die Codierung von MIDI-Nachrichten erläutert.

Byte-Arten

Im MIDI-Format gibt es zwei Arten von Bytes: Statusbytes und Datenbytes [47]. Beim Statusbyte ist das Most significant bit (MSB) auf 1 gesetzt zur Unterscheidung zu Datenbytes. Statusbytes legen die Art der übertragenen Nachricht fest und wie die folgenden Datenbytes zu interpretieren sind [47].

Kanal-Nachrichten

Das MIDI-Format kann Informationen für bis zu 16 sogenannte Kanäle kontinuierlich übertragen [47]. Ein Kanal kann vereinfacht als eine musikalische Stimme oder ein Instrument betrachtet werden. Bei zum Beispiel zwei genutzten Kanälen kann der Kanal 1 eine Gitarre simulieren und Kanal 2 ein Keyboard. Die Kanäle können gleichzeitig Töne ausgeben und sind in ihren Einstellungen bezüglich Klangart, Lautstärke und Tonerzeugungsdynamik völlig unabhängig [47]. Kanal-Nachrichten können entweder Voice- oder Mode-Nachrichten sein, wobei nur ein Ausschnitt der Voice-Nachrichten für das Verständnis der Arbeit relevant sind. Voice-Nachrichten bestehen aus einem Statusbyte und ein oder zwei Datenbytes [47]. Die wichtigsten Voice-Nachrichten sind „Note On“ und „Note Off“ um einen Ton zu erzeugen oder ihn zu beenden. Deren Aufbau ist in Tabelle 2.2 dargestellt. Die Tonnummer im ersten Datenbyte bezieht sich auf die MIDI-Tonnummern, wobei 0 für den Ton C,,, (8,176 Hz) und 127 für den Ton g”” (12543,9 Hz) steht [47]. Die vollständige Zuordnung von Tonnummern zu den deutschen und englischen Tonnamen und der Tonfrequenz ist in Tabelle A.1 im Anhang zu finden. Im MIDI-Standard wird die Bezeichnung Geschwindigkeit im zweiten Datenbyte verwendet, wobei dieser Wert oft als logarithmische Lautstärke interpretiert wird. Eine Besonderheit ist, dass auch eine „Note On“-Nachricht mit der Geschwindigkeit 0 genutzt werden kann, um einen Ton zu beenden [47]. Das wird vor allem für den sogenannten „Running Mode“ der Übertragung verwendet [47]. Im „Running Mode“ wird ein Statusbyte gesendet und es können unendlich viele Datenbytes nachfolgend gesendet werden, die immer mit diesem Statusbyte assoziiert werden, bis ein neues Statusbyte gesendet wird. Dadurch wird die Nachrichtenübertragung kurz gehalten, da sich sonst wiederholende Statusbytes nur einmal gesendet werden. Neben „Note Off“ und „Note On“ gibt es noch die Voice-Nachricht „Program Change“, über die man Klang-Voreinstellungen des jeweiligen MIDI-Ausgabegerätes auswählen kann [47]. Bei einer MIDI-Gitarre kann man darüber einstellen, ob sie zum Beispiel klar

oder stark verzerrt klingen soll. Man kann die Einstellung 0 bis zur Einstellung 127 anwählen, wobei nicht jedes MIDI-Gerät tatsächlich so viele Optionen bietet. In [46] ist eine Belegungstabelle für die 128 Programmplätze vorgeschlagen. Halten MIDI-Geräte die „General MIDI Specifications“ ein, sind die Programmplätze entsprechend belegt. Für den Sound „Electric Guitar (clean)“ wird Programmplatz 28 verwendet. Programmplatz 1 erzeugt einen Konzertflügel-Sound [46]. Wird eine ungültige Nummer übermittelt, wird die Nachricht ignoriert [47]. Einstellung 1 ist die Standard-Einstellung für die meisten Geräte. Der Aufbau der „Program Change“-Nachricht ist in Tabelle 2.2 aufgeführt.

Tabelle 2.2: Aufbau der Voice-Nachrichten „Note Off“, „Note On“ und „Program Change“

Statusbyte	1. Datenbyte	2. Datenbyte	Name: Erklärung
1000nnnn	0kkkkkkk	0vvvvvvv	Note Off: Beendet im Kanal nnnn den Ton kkkkkkk mit der Geschwindigkeit vvvvvvv
1001nnnn	0kkkkkkk	0vvvvvvv	Note On: Erzeugt im Kanal nnnn den Ton kkkkkkk mit der Geschwindigkeit vvvvvvv
1100nnnn	0ppppppp	-	Program Change: Ändert für Kanal nnnn die Einstellung des MIDI-Gerätes auf Programm ppppppp

2.12.3 Dateiformat

Statt MIDI-Geräte direkt im MIDI-Format anzusteuern, können auch MIDI-Dateien eingespielt werden und die MIDI-Geräte interpretieren diese. Zusätzlich zu den normalen MIDI-Nachrichten bestehen MIDI-Dateien aus einem Datei- und mindestens einem Track-Header, die Informationen zum Aufbau der Datei, der Anzahl der Kanäle und der Länge der Zeiteinheiten innerhalb der Datei enthalten [48].

Datei-Header und Track-Header

Der Datei-Header besteht aus:

- Header-Art (4 Byte)
- Anzahl folgender Bytes im Header (4 Byte)

- Formatangabe (2 Byte)
- Anzahl der Tracks in der Datei (2 Byte)
- Angabe über die Einteilung der Zeiteinheiten; oft Angabe, wie viele Zeiteinheiten eine Viertelnote bilden (2 Byte)

Für Datei-Header heißt die Header-Art *MThd* und es folgen immer sechs Bytes im Datei-Header nach der Byteangabe, das heißt der gesamte Datei-Header ist 14 Bytes groß [48]. Für das Format der Datei gibt es drei Optionen: ein einzelner Track (Format 0), der mehrere MIDI-Kanäle beinhalten kann, mehrere Tracks, die gleichzeitig interpretiert werden sollen (Format 1), oder mehrere Tracks, die unabhängig voneinander nacheinander interpretiert werden sollen (Format 2) [48]. Dabei ist ein Track wie eine Tonspur zu verstehen. Für das Einzel-Track-Format gibt es logischerweise nur einen Track in der Datei, für die anderen zwei Formate können bis zu 16 Tracks aufgeführt sein [48].

Im Rahmen dieser Arbeit wird bei der Einteilung der Zeiteinheiten von der Anzahl sogenannter „Ticks“ pro Viertelnote gesprochen.

Nach dem Datei-Header folgen mindestens ein Track-Header samt dazugehörigen Track. Der Track-Header besteht aus:

- Header-Art (4 Byte)
- Anzahl folgender Bytes im gesamten Track (variable Byteanzahl)

Für Tracks heißt die Header-Art *MTrk*. Nach der Längenangabe des gesamten Tracks erfolgt mindestens ein Track-Event.

Track-Events

Track-Events bestehen aus einer Delta-Zeit und einer MIDI-Nachricht, einer System-Nachricht oder einem Meta-Event [48]. Die Delta-Zeit gibt dabei die zeitliche Differenz zum vorherigen Event an. Eine Delta-Zeit von 0 bedeutet, dass die Events gleichzeitig stattfinden [48].

Inhaltlich relevante, aber zum Abspielen der Datei technisch optionale, Meta-Events beinhalten Informationen zum Beispiel zum Urheberrecht, zum Tracknamen, zur musikalischen Zeitsignatur und zum Tempo [48]. Sind die Zeitsignatur und das Tempo nicht

definiert, wird von einem 4/4-Takt bei 120 BPM (Beats Per Minute) als Standard ausgegangen [48]. Diese Meta-Events sollen idealerweise zum Zeitpunkt 0 des Tracks definiert sein, Tempowechsel und Tonartwechsel innerhalb des Tracks ausgenommen. Meta-Events beginnen immer mit dem Byte 0b1111111 (hex. 0xFF) [48]. Zwingend notwendig ist das Meta-Event zum Track-Ende [48].

Meta-Events bestehen aus zwei Bytes für den Event-Typ, einer Angabe zur Anzahl der folgenden Datenbytes und einer variablen Menge von Datenbytes [48]. Wenn Datenbytes Bezeichnungen beinhalten, werden diese in American Standard Code for Information Interchange (ASCII) codiert. Die Meta-Events für Trackname, Instrumentenname, Taktart, Tonart, Tempo und Track-Ende werden im Folgenden erläutert.

Trackname:

- Event-Typ: 0xFF 0x03
- Anzahl folgender Bytes (variable Länge)
- Name des Tracks in ASCII (variable Länge)

Instrumentenname:

- Event-Typ: 0xFF 0x04
- Anzahl folgender Bytes (variable Länge)
- Name des Instruments in ASCII (variable Länge)

Zeitsignatur:

- Event-Typ: 0xFF 0x58
- Anzahl folgender Bytes: 4 (1 Byte)
- Zähler der Zeiteinteilung: (1 Byte)
- Nenner der Zeiteinteilung in negativen Potenzen von 2 (1 Byte)
- Anzahl der Synchronisationsimpulse pro musikalischen Metronomschlag (1 Byte), nicht zu verwechseln mit der Anzahl an Ticks pro Viertelnote

- Anzahl an 32stel-Noten pro Viertelnote, meistens acht (1 Byte)

Tonart:

- Event-Typ: 0xFF 0x59
- Anzahl folgender Bytes: 2 (1 Byte)
- Angabe über die Vorzeichen: -7 = 7 b, 0 = kein Vorzeichen, 7 = 7 # (1 Byte)
- Angabe über das Tongeschlecht: 0 = Moll, 1 = Dur (1 Byte)

Tempoangabe:

- Event-Typ: 0xFF 0x51
- Anzahl folgender Bytes: 3 (1 Byte)
- Länge einer Viertelnote in μs (3 Byte)

Track-Ende:

- Event Typ: 0xFF 0x2F 0x00

Zeitangaben

Die Angabe der Delta-Zeit in MIDI-Dateien erfolgt mit einer speziellen Codierung namens Variable Längenangabe (VLA) [48]. Die Delta-Zeit wird in Vielfachen der Zeiteinheit „Ticks“ dargestellt. Es werden maximal vier Bytes zur Darstellung verwendet [48]. Die größte darstellbare Zahl ist dabei 0xFFFFFFFF. Zwischen 0 und 127 Ticks wird die Delta-Zeit normal binär in einem Byte dargestellt [48]. Ab 128 aufwärts, wenn das MSB des Byte gesetzt werden müsste, wird das MSB auf 0 gesetzt. Es wird ein weiteres Byte vorangestellt, dessen sieben LSB den nächsten sieben Bits der Ausgangszahl entsprechen. Das MSB dieses vorangestellten Bytes wird auf 1 gesetzt. Bei Bedarf wird so lange ein weiteres Byte mit gesetztem MSB vorangestellt, bis alle Bits der Ausgangszahl durch die jeweils sieben LSB der VLA dargestellt wurden. Beispiele für die VLA sind in Tabelle 2.3 zu sehen.

Tabelle 2.3: Beispiele für die variable Längenangabe der Delta-Zeit

binäre Zahl	hexadezimale Zahl	binäre VLA	hexadezimale VLA
0b0111111	0x7F	01111111	0x7F
0b00000001.11100000	0x01.E0	0b10000011.01100000	0x83 0x60
0b00111111.11111111	0x3F.FF	0b1111111.01111111	0xFF 0x7F

2.12.4 Minimalbeispiel für den Aufbau einer MIDI-Datei

MIDI-Dateien sind binär codiert. Ihr Inhalt steht ohne Leerzeichen in einer Zeile geschrieben. Für das folgende Beispiel in der Tabelle 2.4 sind der Lesbarkeit halber Zeilenumbrüche inkludiert und die Zeichen hexadezimal dargestellt. Die verwendete Tonfolge ist in Abbildung 2.20 dargestellt.

A musical staff in 4/4 time with a treble clef. The tempo is indicated as $\text{♩} = 120$. The notes are represented by vertical stems with small circles at the top. Below the staff, the notes are labeled with lowercase letters: E, A, d, g, h, e'. The note 'e'' is positioned above the staff, indicating it is the next note in the sequence.

Abbildung 2.20: Beispiel-Tonabfolge

Tabelle 2.4: Minimalbeispiel einer MIDI-Datei

Delta-Zeit	Inhalt/Event	Erklärung
-	0x4D 0x54 0x68 0x64	Datei-Header „MThd“
-	0x00 0x00 0x00 0x06	Es folgen noch 6 Bytes im Datei-Header
-	0x00 0x00	Einzel-Track-Format
-	0x00 0x01	Ein Track in der Datei
-	0x01 0xE0	480 Ticks pro Viertelnote
-	0x4D 0x54 0x72 0x6B	Track-Header „MTrk“
-	0x00 0x00 0x00 0x48	Insgesamt 68 Bytes bis zum Track-Ende
0x00	0xFF 0x03 0x07 0x47 0x69 0x74 0x61 0x72 0x72 0x65	Track-Name ist 7 Byte lang und lautet Gitarre
0x00	0xFF 0x51 0x03 0x07 0xA1 0x20	Eine Viertelnote ist 500 ms lang (120 BPM)
0x00	0xC0 0x00	Standard-Sound 0 verwenden
0x00	0x90	Ton in Kanal 0 erzeugen
-	0x28 0x50	Ton E mit Lautstärke 50 spielen
0x83 0x60	0x28 0x00	Ton E mit Lautstärke 0 beenden
0x00	0x2D 0x50	Ton A mit Lautstärke 50 spielen
0x83 0x60	0x2D 0x00	Ton A mit Lautstärke 0 beenden
0x00	0x32 0x50	Ton d mit Lautstärke 50 spielen
0x83 0x60	0x32 0x00	Ton d mit Lautstärke 0 beenden
0x00	0x37 0x50	Ton g mit Lautstärke 50 spielen
0x83 0x60	0x37 0x00	Ton g mit Lautstärke 0 beenden
0x00	0x3B 0x50	Ton h mit Lautstärke 50 spielen
0x83 0x60	0x3B 0x00	Ton h mit Lautstärke 0 beenden
0x00	0x40 0x50	Ton e' mit Lautstärke 50 spielen
0x83 0x60	0x40 0x00	Ton e' mit Lautstärke 0 beenden
0x00	0xFF 0x2F 0x00	Track-Ende

3 Stand der Technik

In diesem Kapitel wird eine Auswahl hexaphonischer und MIDI-Tonabnehmer vorgestellt, die zum Zeitpunkt der Erstellung dieser Arbeit frei erwerbbar sind und den Stand der Technik auf diesen Gebieten abbilden. In diesem Kapitel steht der Begriff „Hexaphonischer Tonabnehmer“, im Gegensatz zu MIDI-Tonabnehmern, für Tonabnehmer, deren Einsatzbereich nicht ausschließlich auf die MIDI-Konvertierung beschränkt ist. Denn MIDI-Tonabnehmer sind grundsätzlich immer polyphon aufgebaut. Anschließend werden die aktuell für Gitarrenmusik verfügbaren MIDI-Konverter vorgestellt, die in Form einer Softwareanwendung aufgenommene Töne ohne hexaphonischen Tonabnehmer in das MIDI-Format übersetzen können. Zuletzt werden die zum Zeitpunkt der Erstellung dieser Arbeit relevanten Algorithmen und Verfahren zur Tonhöhenerkennung vorgestellt.

3.1 Hexaphonische Tonabnehmer

GK-3 ist ein hexaphonischer Tonabnehmer inklusive Controller mit aktiver Verstärkung der Firma Roland, der zusätzlich zu vorhandenen Tonabnehmern an E-Gitarren nachgerüstet werden kann [41]. An der Controllereinheit kann der Ausgabepegel eingestellt werden und über eine zusätzliche Klinkenbuchse das monophone Ausgangssignal der E-Gitarre angeschlossen werden. Über das 13-polige „GK-Interface“ kann der Tonabnehmer mit Gitarrensynthesizern oder MIDI-Interfaces des gleichen Herstellers angeschlossen werden. Über diese Verbindung erhält der Controller auch seine Versorgungsspannung. Zum internen Aufbau des Tonabnehmers selbst gibt es keine Angaben, lediglich der Schaltplan der Controllereinheit ist veröffentlicht. Zum Zeitpunkt der Erstellung dieser Arbeit liegt der Kaufpreis für einen „GK-3“ bei 199 €¹.

¹https://www.thomann.de/de/roland_gk3_midipickup.htm

NU Series Active Modular Pickup ist ein Open Source entwickelter hexaphonischer Tonabnehmer der Firma Cycfi mit aktiver Verstärkung [5]. Die Daten zum Aufbau des Tonabnehmers inklusive Informationen zur Verstärkerschaltung und Aufbau der Montageplatine sind frei einsehbar. Die Spulen sind mit Kupferlackdraht der Dicke American Wire Gauge (AWG) 46 (Durchmesser: 0,03988 mm) mit 1.300 Wicklungen gewickelt. Der Drahtwiderstand beträgt 400Ω . Die Übertragungsfunktion des Tonabnehmers ist über den Frequenzbereich 20 Hz bis 20 kHz nahezu linear. In der Basisplatine der Spulen befindet sich eine Emitterschaltung zur Verstärkung der Spannung auf den Bereich ± 374 mV bei einer Ausgangsimpedanz von $10 \text{ k}\Omega$. Als Spannungsversorgung ist eine Spannung im Bereich von 5 V bis 18 V zulässig. Den Tonabnehmer in Single Coil-Montageform kann man fertig zur Montage in E-Gitarren mit sechs Saiten kaufen. Es gibt aber auch einzelne Spulen für eine individuelle Anpassung an mehr oder weniger Saiten. Über eine zusätzliche Anschlussbox ist es möglich, den Tonabnehmer über ein Kabel mit 13-poligen DIN-Stecker an Synthesizer und MIDI-Interfaces mit „GK-Interface“ der Firma Roland anzuschließen. Zum Zeitpunkt der Erstellung dieser Arbeit liegt der Kaufpreis für einen „NU Multi 6“ in Standardausführung bei 189,99 \$².

Ubertar Hexaphonic Pickup ist ein hexaphonischer Tonabnehmer, den es in Single Coil- als auch in Humbuckerbauform gibt [34]. Der Tonabnehmer wird gegen einen vorhandenen monophonen Tonabnehmer in der E-Gitarre ausgetauscht. Im Gegensatz zum „NU Multi 6“ wird keine aktive Verstärkung und zusätzliche Spannungsquelle benötigt. Die Spulen haben einen Drahtwiderstand von $3 \text{ k}\Omega$. Zusätzlich zum Tonabnehmer gibt es auch hier eine Anschlussbox, von der sechs Klinkenbuchsen für die sechs Tonsignale abgehen. Da bei diesem Hersteller der Tonabnehmer nicht fertig konfektioniert mit einem Kabel samt Anschlussstecker geliefert wird, sondern die einzelnen Adern des Tonabnehmers selbst an einen Stecker gelötet werden müssen, kann auch ein 13-poliger DIN-Stecker verwendet werden zum Anschluss an Roland-Geräte. Zum Zeitpunkt der Erstellung dieser Arbeit liegt der Kaufpreis für einen „Ubertar Hexaphonic Pickup“ im Single Coil-Format bei 165 \$ und für eine Abschlussbox bei 85 \$³.

ghost Pickup ist ein hexaphonischer Piezo-Tonabnehmer, der als Stegtonabnehmer eingebaut wird [18]. Über eine zusätzliche Verstärkerplatine können die Signale abgegriffen werden. An der Platine kann ein Kabel mit 13-poligen DIN-Stecker angeschlossen

²<https://www.cycfi.com/product/nu-multi-6/>

³<https://ubertar.com/hexaphonic/products.html>

werden, um den Tonabnehmer mit Roland-Geräten zu verwenden. Zum Zeitpunkt der Erstellung dieser Arbeit liegt der Kaufpreis für einen „ghost Pickup“ zwischen 126,04 € und 231,95 €⁴ und für eine Verstärkerplatine bei 141,57 €⁵.

3.2 MIDI-Tonabnehmer

Triple Play Wireless ist ein hexaphonischer Tonabnehmer mit MIDI-Controller, der zusätzlich zu vorhandenen Tonabnehmern an E-Gitarren nachgerüstet werden kann [13]. Der Tonabnehmer ist fest am Controller angeschlossen, sodass auf die einzelnen Rohsignale von außen nicht zugegriffen werden kann. Der MIDI-Controller sendet seine Daten an ein Empfangsgerät, das am Computer angeschlossen ist. Die Reichweite liegt hierbei bei 30 m und die Akku-Laufzeit beträgt 20 Stunden. Alternativ gibt es auch eine kabelgebundene Ausführung. Zum Zeitpunkt der Erstellung dieser Arbeit liegt der Kaufpreis für einen „Triple Play Wireless“ bei 379 €⁶.

3.3 MIDI-Interfaces

Mitte der 90er kamen MIDI-Interfaces von Roland, Yamaha und Blue Chip auf den Markt, die mithilfe hexaphonischer Tonabnehmer Gitarrentöne in das MIDI-Format übersetzten und diese Daten entweder an einen Computer oder andere MIDI-Geräte weitergeleitet haben [50, 51, 28]. Die Geräte von Yamaha und Blue Chip setzen neuronale Netze zur Tonhöhenanalyse ein [28]. Aktuell werden diese Geräte nicht mehr produziert und sind nur noch gebraucht erhältlich. Die hexaphonischen Tonabnehmer werden weiterhin zum Einsatz mit Gitarren-Synthesizern produziert, um die gespielten Töne mit Effekten zu bearbeiten. Inzwischen haben sich Softwareanwendungen zur reinen MIDI-Konvertierung durchgesetzt.

⁴<https://graphtech.com/collections/ghost-pickup-systems-guitar-saddle>

⁵<https://graphtech.com/collections/ghost-pickup-systems-parts-accessories/products/ghost-hexpander-preamp-board>

⁶https://www.thomann.de/de/fishman_triple_play.htm

3.4 MIDI-Konverter

MiGiC ist eine Software um mit geringer Latenz Gitarrentöne in das MIDI-Format zu übersetzen und direkt mit digitalen Effekten bearbeitet auszugeben [32]. Die Software kann aktuell nur monophone Signale übersetzen, benötigt dafür aber auch keinen hexaphonischen Tonabnehmer. Eine Angabe zur genauen Latenz der Konvertierung oder dem Ansatz zur Tonhöhenerkennung ist nicht vorhanden. Die Software kann sowohl als Erweiterung für bestehende Digital Audio Workstation (DAW)-Programme verwendet werden als auch alleine. Zum Zeitpunkt der Erstellung dieser Arbeit kostet eine Lizenz für „MiGiC“ 29,95 \$⁷.

MIDI Guitar 2 ist eine Software um mit geringer Latenz selbst polyphone Gitarrenmusik in das MIDI-Format zu übersetzen und direkt mit digitalen Effekten bearbeitet auszugeben [23]. Für die polyphonen Tonhöhenerkennung wird kein zusätzlicher hexaphonischer Tonabnehmer benötigt. Statt klassischer Signalverarbeitung werden Deep Learning-Techniken und künstliche Intelligenz eingesetzt [52]. Die Software kann sowohl als Erweiterung für bestehende DAW-Programme verwendet werden als auch alleine. Zum Zeitpunkt der Erstellung dieser Arbeit kostet eine Lizenz für „MIDI Guitar 2“ 149,95 \$⁸.

3.5 Pitch Detection Algorithmen

Die Ansätze von PDA lassen sich in die drei Kategorien Zeitbereich, Frequenzbereich und maschinelles Lernen und neuronale Netze unterteilen. Im Folgenden werden sowohl aktuelle als auch altbewährte PDA der jeweiligen Kategorie vorgestellt.

3.5.1 Ereignisse im Zeitbereich

Abstand der Nulldurchgänge

Ein intuitiver Ansatz zur Bestimmung der Grundperiode T_0 eines Tones ist die Auswertung der Nulldurchgänge, der in [24] vorgestellt wurde. Es gilt die Annahme, dass eine periodische Schwingung zwei Nulldurchgänge in jeder Periode aufweist, sodass aus dem

⁷<https://migic.com/purchase>

⁸<https://www.jamorigin.com/products/>

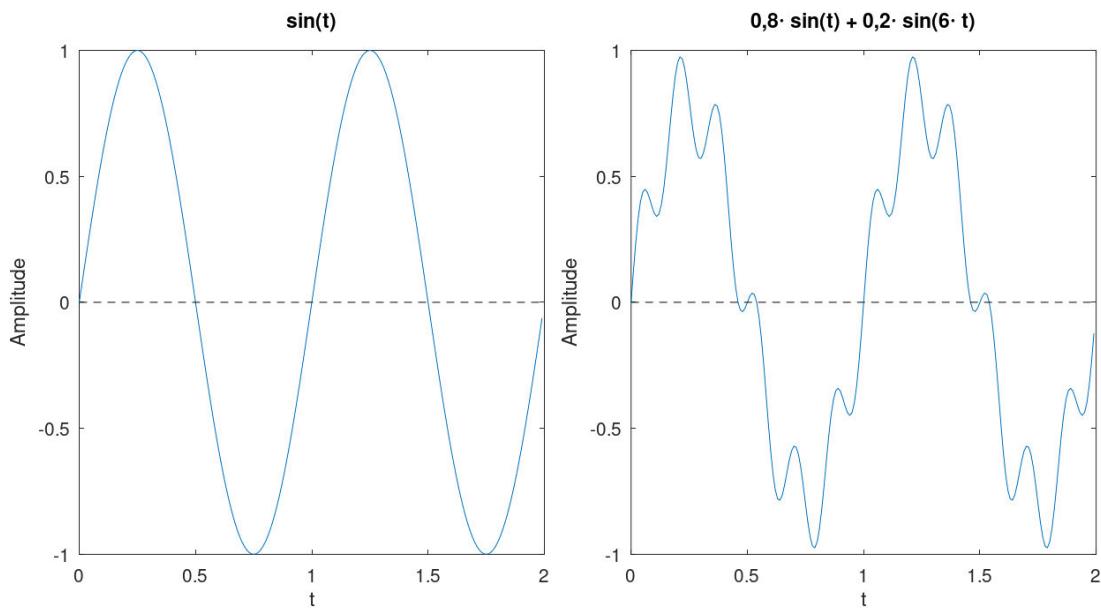


Abbildung 3.1: Einfluss von Obertönen auf die Anzahl der Nulldurchgänge

zeitlichen Abstand der Nulldurchgänge die Signalperiode bestimmt werden kann, siehe Abbildung 3.1 für $\sin(t)$. Für Tongemische mit Obertönen kommt das Verfahren schnell an seine Grenzen, wie man in Abbildung 3.1 für $0,8 \sin(t) + 0,2 \sin(6t)$ sehen kann [39]. Dort treten pro Signalperiode vier statt zwei Nulldurchgänge auf.

Abstand der Maxima und Minima

Analog zum Abstand der Nulldurchgänge kann auch der Abstand der Maxima oder Minima betrachtet werden [16]. Je nach Zusammensetzung des Tons können wiederum mehrere nahezu gleich große Maxima beziehungsweise Minima auftreten, die die Auswertung erschweren.

3.5.2 Autokorrelation

Bei der Autokorrelation wird ein Zeitsignal mit sich selbst, verzögert um einen zeitlichen Abstand τ , multipliziert. Sie ist ein Maß für die Ähnlichkeit des Signals und des

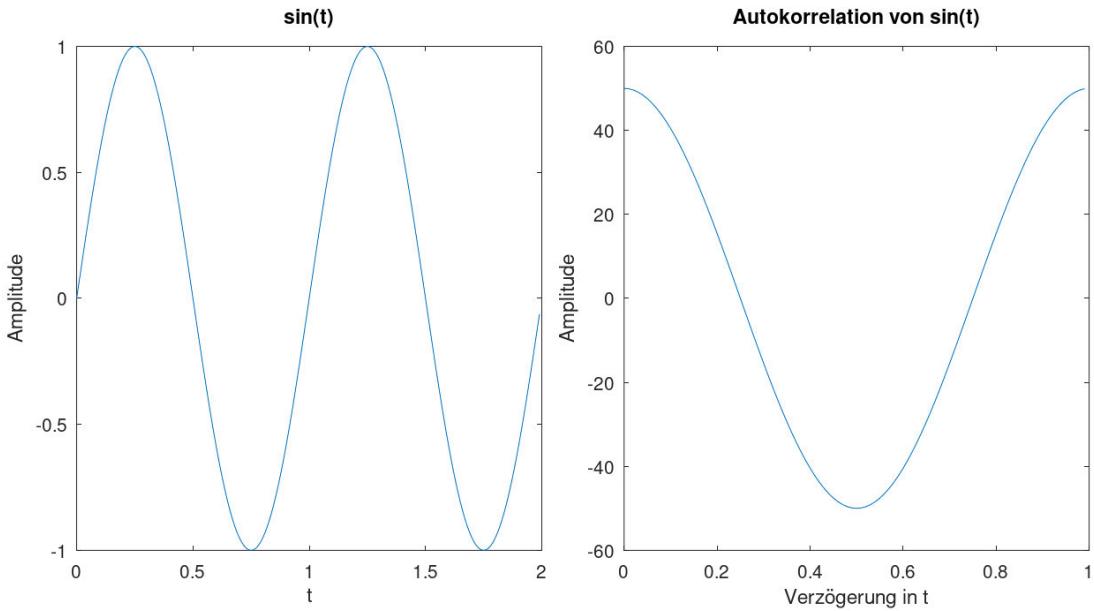


Abbildung 3.2: Verlauf der Autokorrelation einer Sinusschwingung

verzögerten Signals. Für diskrete Zeitsignale kann die Autokorrelation für ein endliches Zeitfenster N wie folgt beschrieben werden [16]:

$$R_{xx}(\tau) = \sum_{n=0}^{N-1-\tau} x(n) \cdot x(n + \tau) \quad (3.1)$$

Für eine Verzögerung von $\tau = 0$ ist die Autokorrelation maximal, da ein Signal um 0 verzögert sich selbst entspricht. Periodische Signale haben auch eine periodische Autokorrelation [16]. Für eine einfache Sinusschwingung mit nur einer Grundfrequenz ist die Autokorrelation für eine Verzögerung um die Hälfte der Periodenlänge minimal und bei einer Verzögerung um eine Periodenlänge wieder maximal, siehe Abbildung 3.2. Problematisch wird es, wenn nicht-ideale Schwingungen mit abklingender Amplitude betrachtet werden, da die Autokorrelation stark von der absoluten Amplitude der Schwingung abhängt. Bei einer Verschiebung um eine Periodenlänge wird sich bei der Autokorrelation zwar ein lokales Maximum ausbilden, dieses wird aber nicht den gleichen Wert haben wie das globale Maximum für eine Verzögerung um 0 [16].

YIN

Der YIN-Algorithmus wurde 2002 veröffentlicht und ist seitdem immer noch einer der bekanntesten PDA. Der Name, abgeleitet von den Begriffen „Yin“ und „Yang“ aus der chinesischen Philosophie, spielt auf das gegensätzliche Verhältnis aus Autokorrelation und Auslöschung an, die beide Bestandteile des Algorithmus sind. Im Vergleich zu später entwickelten PDA weist YIN zwar eine höhere Latenz aber dafür die geringste Fehlerrate bei der korrekten Bestimmung von f_0 auf [1, 37]. Eine Implementierung in MATLAB ist unter [7] zu finden.

Verfahren: Beim YIN-Algorithmus wird im Gegensatz zur normalen Autokorrelation nicht das Maximum des Produktes aus Ausgangssignal und verschobenem Signal gesucht sondern das Minimum aus deren Differenz [8]:

$$d_t(\tau) = \sum_{j=1}^W (x_j - x_{j+\tau})^2 \quad (3.2)$$

$d_t(\tau)$ hat bei $\tau = 0$ das globale Minimum von 0 und für Vielfache der gesuchten Periodenlänge Werte nahe bei 0, da bei nicht-idealen Schwingungen leichte Unterschiede auftreten, die sich bei der Differenz nicht vollständig aufheben. Um zu verhindern, dass $\tau = 0$ oder Periodenlängen von Obertönen mit kleineren Differenzen als T_0 ausgewählt werden, wird $d_t(\tau)$ durch seinen kumulierten Mittelwert dividiert. Diese neue Funktion $d'_t(\tau)$ wird „cumulative mean normalized difference function“ genannt [8]. $d'_t(0)$ wird fest auf 1 gesetzt, um als mögliches Minimum für T_0 ausgeschlossen zu werden.

$$d'_t(\tau) = \begin{cases} 1 & \text{für } \tau = 0 \\ \frac{d_t(\tau)}{\left[\frac{1}{\tau} \sum_{j=1}^{\tau} d_t(j) \right]} & \text{sonst} \end{cases} \quad (3.3)$$

Es wird ein Schwellwert festgelegt, unter dem $d'_t(\tau)$ liegen muss, damit τ für T_0 ausgewählt wird. Anschließend wird ab $\tau = 1$ das erste Minimum von $d'_t(\tau)$ unter diesem Schwellwert gesucht und die Suche beendet. Liegt $d'_t(\tau)$ vollständig über dem Schwellwert, wird das globale Minimum gewählt. Die Funktionswerte von $d'_t(\tau)$ sind als Maß der Aperiodizität zu interpretieren, das heißt je niedriger $d'_t(\tau)$ ist, desto eher ist die Verschiebung um τ ein periodisches Vielfaches des Ausgangssignals.

Bei geringen Abtastfrequenzen, respektive großen zeitlichen Abständen zwischen den

Datenpunkten, kann es passieren, dass die gesuchte Periode zwischen zwei Verzögerungswerten liegt. Um diesen Fehler zu reduzieren, werden das ermittelte τ und seine zwei Nachbarn parabolisch interpoliert und das Interpolationsminimum als T_0 betrachtet. Zuletzt werden die vorherigen Schritte für ein kleines Intervall $[t - T_{max}/2, t + T_{max}/2]$ um t herum wiederholt und die T_0 -Schätzung mit der geringsten Aperiodizität wird als finales Ergebnis zum Zeitpunkt t verwendet. T_{max} ist dabei die größte erwartete Grundperiode. Diese Suche reduziert Fehler im Abhängigkeit der Signalphase.

Randbedingungen: Der YIN-Algorithmus kann nur monophone Signale analysieren und hat eine minimale Latenz von $T_{max} + T_0$. Das liegt daran, dass nach der Berechnung von $d'_t(\tau)$ die Suche nach dem Minimum nach spätestens $\tau = T_0$ beendet wird [8]. Unabhängig davon müssen vor der Berechnung initial mindestens $2 \cdot T_{max}$ an Datenpunkten aufgenommen werden. Bei Verfahren basierend auf der Autokorrelation können sogenannte Oktavfehler auftreten, wenn Obertöne mit geringerem T_0 fälschlicherweise als Grundton erkannt werden. Das liegt daran, dass die Verschiebung um die Periodenlänge eines Obertons das erste gültige lokale Minimum erzeugen kann, obwohl der Extremwert für das korrekte T_0 besser wäre, aber im Algorithmus nicht erreicht wird.

pYIN

Der probabilistische YIN (pYIN)-Algorithmus aus 2014 ist eine Erweiterung des YIN-Algorithmus [31]. Es wird angenommen, dass durch Variation des Schwellwerts zur Bestimmung des ersten Minimums im YIN-Algorithmus die Ergebnisse für T_0 verbessert werden können. pYIN weist im Vergleich zu YIN sowohl eine höhere Sensitivität als auch eine höhere Spezifität auf [31]. In [30] dient pYIN als Grundlage für eine Tonhöhenerkennungs- und Musiknotationssoftware. In der Nutzungsoberfläche werden die erkannten Tonhöhen angezeigt, Nutzende können Korrekturen vornehmen und die Ergebnisse als Comma-separated values (CSV)-Datei exportieren.

Verfahren: Im ersten Schritt wird der YIN-Algorithmus so erweitert, dass er nicht einen T_0 -Kandidaten ausgibt, sondern mehrere für verschiedene Schwellwerte. Im zweiten Schritt wird mit einem Hidden Markov Model anhand der T_0 -Kandidaten und deren Aperiodizität der finale T_0 -Wert bestimmt.

Randbedingungen: Auch pYIN kann wie YIN nur monophone Signale analysieren.

OneBitPitch

OneBitPitch basiert wie YIN auf der Differenzform der Autokorrelation, ist aber auf schnelle Rechenzeit und weniger akkurate Ergebnisse ausgelegt [4].

Verfahren: Die Daten werden vor der eigentlichen Analyse mit einem Tiefpass gefiltert und die Auflösung auf ein Bit reduziert. Dadurch sind die Datenpunkte entweder 0 oder 1, wobei die Periodizität beibehalten wird. Die Berechnung der einzelnen Differenzen lassen sich dadurch auf eine XOR-Operation reduzieren, die geringe Rechenzeit in Anspruch nimmt.

Randbedingungen: OneBitpitch kann wie YIN nur monophone Signale analysieren. Er ist bis zu neunmal schneller als YIN mit einer durchschnittlichen Programmlaufzeit von 4,2 ms. Der Testdatensatz umfasste 9.600 Samples und die Ausführung erfolgte unter Verwendung von MATLAB Version R2023a auf einem Dell Latitude E5550 Computer mit einem Intel Core i5 5200U Prozessor und einem 16 GB Dual-Channel DDR3 RAM. Die Fehlerquote liegt mit 2 % in einem tolerierbaren Rahmen.

3.5.3 Frequenzbereich

Im Frequenzbereich kann man über eine Fourier-Transformation des Zeitsignals die enthaltenen Frequenzen und deren Anteil an der Gesamtamplitude ermitteln. Intuitiv ist die niedrigste Frequenz als f_0 zu wählen, wobei starke Oberschwingungen die Amplitude von f_0 übersteigen können und die Auswahl erschweren.

Bestimmung der enthaltenen Frequenzen und deren Teilungsverhältnis

Verfahren: In [36] werden aus den erkannten Frequenzen der Fast Fourier Transformation (FFT) Paare gebildet und deren Größter gemeinsamer Teiler (ggT) ermittelt. Dieser Wert dient als Annahme für f_0 , da Obertonfrequenzen ganzzahlige Vielfache von f_0 sind. Nachdem für jedes Frequenzpaar der ggT gebildet wurde, wird per Mehrheitsentscheid

aus den ggT f_0 ausgewählt, wobei die Gewichtung der f_0 -Kandidaten von der Amplitude der Frequenzpaare abhängt. Je höher die Amplituden, desto stärker wird deren ggT gewichtet.

Randbedingungen: Diese Analyse der Obertonfrequenzen und ihrer ggT kann nur für monophone Signale angewandt werden, da der ggT von Obertonfrequenzen verschiedener Grundfrequenzen zu einer dritten vermeintlichen Grundfrequenz führt, die nicht im Signal enthalten ist.

Cepstrum-Alanalyse

Der Name „cepstrum“ entsteht, wenn man die ersten vier Buchstaben des Wortes „spectrum“ in umgekehrter Reihenfolge anordnet. Nach dem selben Schema wird in [3] im Kontext von Cepstrum von Alanalyse statt Analyse gesprochen. Die Cepstrum-Alanalyse ist zwar schneller als der YIN-Algorithmus, aber dafür sind die ermittelten Werte für f_0 um bis zu 8 % schlechter [37].

Verfahren: Bei der Cepstrum-Alanalyse wird von einem Zeitsignal die Fouriertransformation durchgeführt, von diesem resultierenden Frequenzspektrum der dekadische Logarithmus gebildet und davon letztendlich noch eine Fouriertransformation durchgeführt [3]. Durch die zweifach angewandte Fouriertransformation wird das Cepstrum nicht über der Frequenz aufgetragen, sondern über der Größe Quefrenz, die der Größe der Zeit entspricht [39]. Bei der Cepstrum-Alanalyse als Ansatz zur Tonhöhenerkennung gilt erneut die Annahme, dass ein Spektrum eines Tones samt seiner Obertöne aus regelmäßig auftretenden Frequenzspitzen mit hohen Amplituden besteht. Der dekadische Logarithmus des Spektrums reduziert die hohen Amplitudenspitzen und das Spektrum liegt mit einem kleineren Wertebereich vor. Da Obertonfrequenzen ganzzahlige Vielfache von f_0 bilden, kann das logarithmierte Spektrum mit einer periodischen Signalform beschrieben werden. Die Periodenlänge ist dabei der Abstand zwischen zwei Frequenzspitzen und hängt mit f_0 zusammen. Bildet man von dem logarithmierten Spektrum die Fouriertransformierte, hat das resultierende Cepstrum ein Maximum bei der Quefrenz, die der Periodenlänge zwischen den Frequenzspitzen im Spektrum entspricht. Dieser Wert ist dann T_0 . Die Einzelschritte der Cepstrum-Alanalyse sind in Abbildung 3.3 dargestellt.

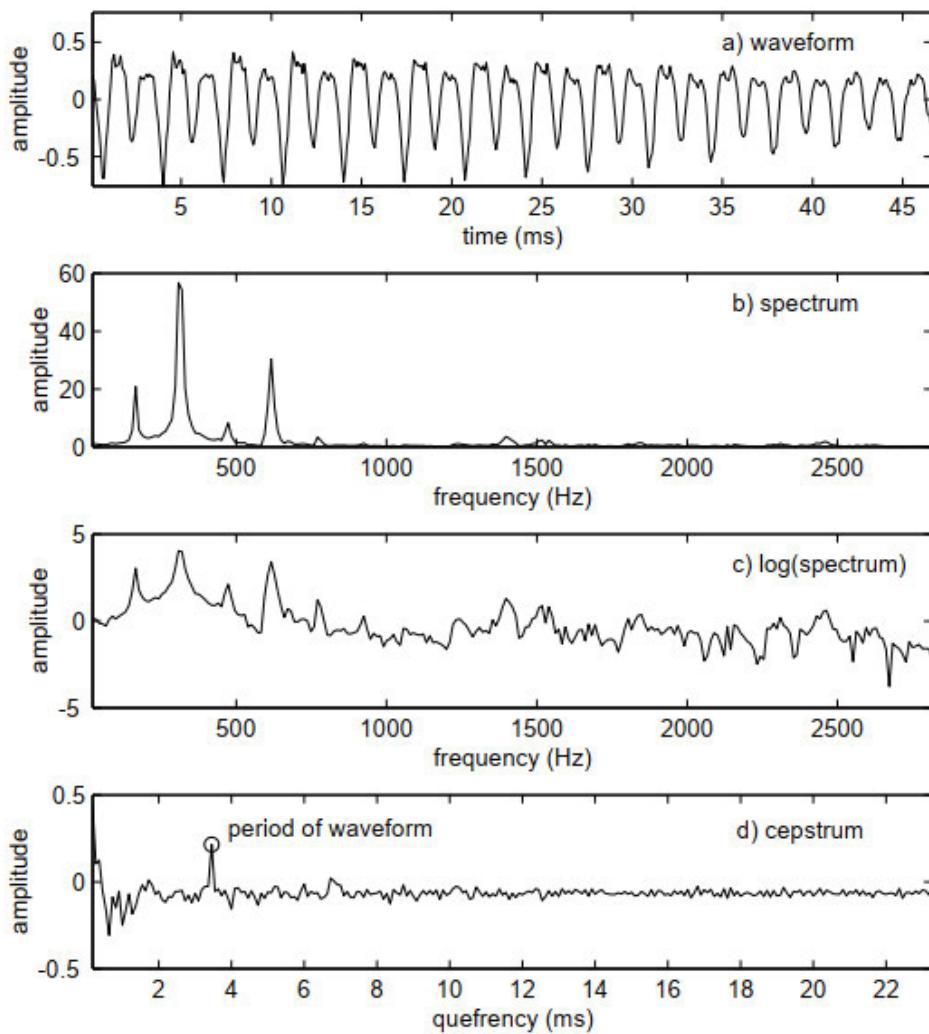


Abbildung 3.3: Schritte der Cepstrum-Alanalyse [16]

Randbedingungen: Für die Cepstrum-Alanalyse ist es wichtig, dass f_0 den größten spektralen Anteil besitzt, da sonst Obertöne als f_0 angenommen werden. Außerdem kann sie nur für monophone Signale angewendet werden.

3.5.4 Maschinelles Lernen und neuronale Netze

Ab der Mitte der 2010er sind immer mehr PDA entwickelt worden, die maschinelles Lernen und neuronale Netze verwenden [10]. Für das überwachte Lernen werden typischerweise Merkmale aus dem Zeit- und Frequenzbereich als Eingangsgrößen kombiniert.

TENT

TENT ist ein PDA der speziell für monophone Gitarrenmusik entwickelt wurde. Er kann sowohl Tonhöhen als auch Spieltechniken, siehe Kapitel 2.4, erkennen. Die Tonhöhenerkennung erfolgt durch bestehende PDA und die Erkennung der Spieltechniken durch Entscheidungsregeln und ein neuronales Netz. Die Ergebnisse beider Analysen werden am Ende zusammengefasst und korrigieren sich gegenseitig.

Verfahren: Im ersten Schritt erfolgt die Tonhöhen- und längenerkennung basierend auf [35], [42] und [31], wobei [42] in Kombination der folgenden Analyseschritte die besten Ergebnisse erzielt. Als nächstes werden die Tonhöhenkonturen aufgeteilt in Bereiche, die nur gespielte Töne enthalten, und in Bereiche, die Spieltechniken enthalten. Die Bereiche mit vermuteten Spieltechniken werden dann anhand fester Entscheidungsmuster und einem neuronalen Netz klassifiziert. Abschließend werden die vorläufigen Tonnamen mithilfe der ermittelten Spieltechniken korrigiert und ergänzt.

Randbedingungen: TENT kann nur monophone Gitarrenmusik analysieren und benötigt mit einer 2.4 GHz Intel Core i5 CPU in Python ca. 1,2 s um eine 4 min lange Datei zu analysieren. Laut der Autorenschaft ist mit Optimierung eine echtzeitfähige Analyse möglich.

Basic Pitch

Basic Pitch ist ein kostenloser Open Source MIDI-Konverter basierend auf [2]. Es können Audiodateien hochgeladen werden und die resultierenden MIDI-Dateien heruntergeladen werden. Basic Pitch kann sowohl monophone als auch polyphone Musik analysieren und kann für jedes Instrument sowie Gesang verwendet werden. Außerdem kann es im Bereich Spieltechniken Bending erkennen. Basic Pitch basiert vollständig auf einem neuronalen

Netz und benötigt als Endanwendung weniger als 20 MB Speicherplatz. Im Bereich monophoner Gitarrenmusik erzielt Basic Pitch bessere Ergebnisse als TENT.

4 Anforderungsanalyse

In diesem Kapitel wird zunächst der Systemaufbau erläutert und daraus der Systemkontext abgeleitet. Dabei wird abgegrenzt, welche Bestandteile des Systems entwickelt werden müssen und welche Schnittstellen zu beachten sind. Danach werden die beteiligten Interessensgruppen der Plattform und ihre Anforderungen betrachtet. Daraus ergeben sich die typischen Anwendungsfälle. Zuletzt werden die Anforderungen der Interessensgruppen durch die Erkenntnisse aus den Anwendungsfällen ergänzt und zusammengefasst.

4.1 Systemaufbau

Das System, vollständig in Abbildung 4.1 zu sehen, besteht aus einer E-Gitarre mit einem hexaphonischen Tonabnehmer, einem Instrumentenkabel, einer Vorverarbeitung und der Konvertierung in das MIDI-Format. Im Folgenden werden die einzelnen Komponenten und ihre Bestandteile näher erläutert.

Hexaphonischer Tonabnehmer: Der hexaphonische Tonabnehmer besteht aus sechs Spulen mit Magnetkern, einem Gehäuse und einer Montageplatte. Der Tonabnehmer wird in der E-Gitarre montiert.

Instrumentenkabel und Anschlusskontakte: Das Instrumentenkabel besteht aus sechs Signal- und sechs Masseverbindungen. Die Anschlusskontakte werden in einer Übergabeeinheit zusammengefasst, die an der E-Gitarre angebracht werden kann. Das Instrumentenkabel wird dann an der Übergabeeinheit angeschlossen.

Vorverarbeitung: Die Vorverarbeitung besteht aus einer Vorverstärkung und der Analog-Digital-Umsetzung der Tonsignale.

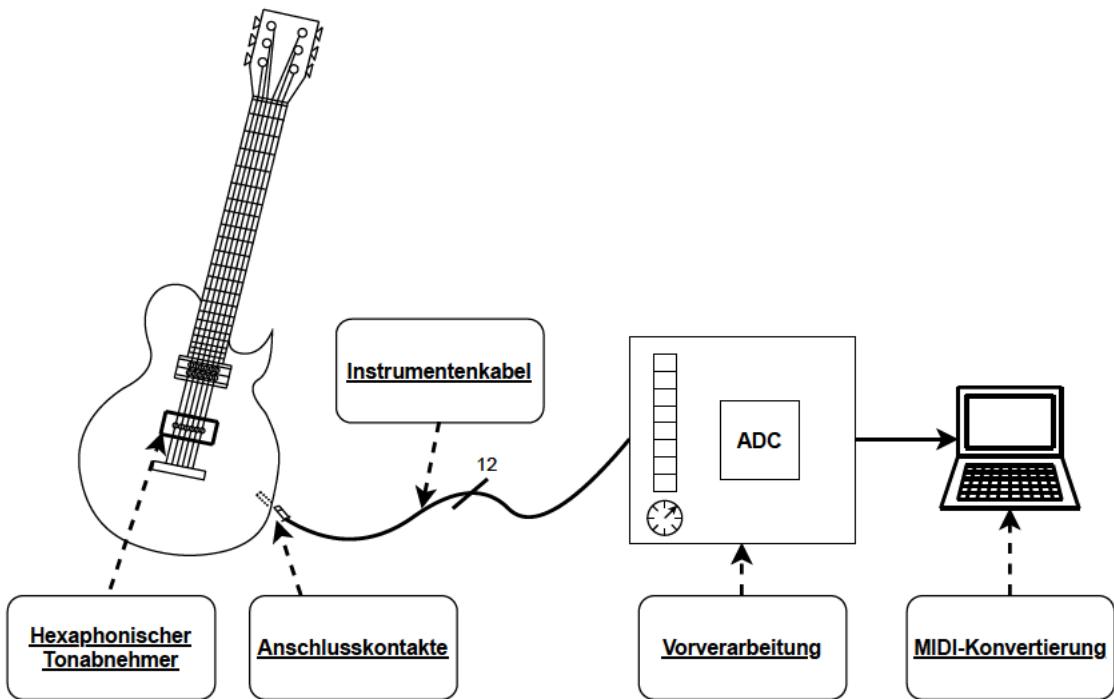


Abbildung 4.1: Systemaufbau

MIDI-Konvertierung: Die MIDI-Konvertierung findet per Software auf einem PC statt. Dafür werden die verstärkten und digitalisierten Signale nach der Vorverarbeitung an den PC übertragen.

4.2 Systemkontext

Der Systemkontext definiert, in welcher Umgebung das System arbeiten wird und welche Bestandteile der Umgebung einen Einfluss auf das System haben können. Die Systemgrenze definiert hierbei, welche Komponenten des Systems während der Entwicklung beeinflussbar sind und welche Anteile der Umgebung außerhalb des Systems liegen und nicht beeinflussbar sind. Die Umgebung ist über die Systemschnittstellen mit dem System verbunden. In Abbildung 4.2 ist der Systemkontext des Systems dargestellt. Im Rahmen dieser Arbeit muss die komplette Verarbeitungskette vom hexaphonischen Tonabnehmer bis zur MIDI-Konvertierung aufgebaut werden, wobei der Schwerpunkt der Selbstentwicklung auf dem Bau des hexaphonischen Tonabnehmers und der Entwicklung der MIDI-Konvertierung liegt.

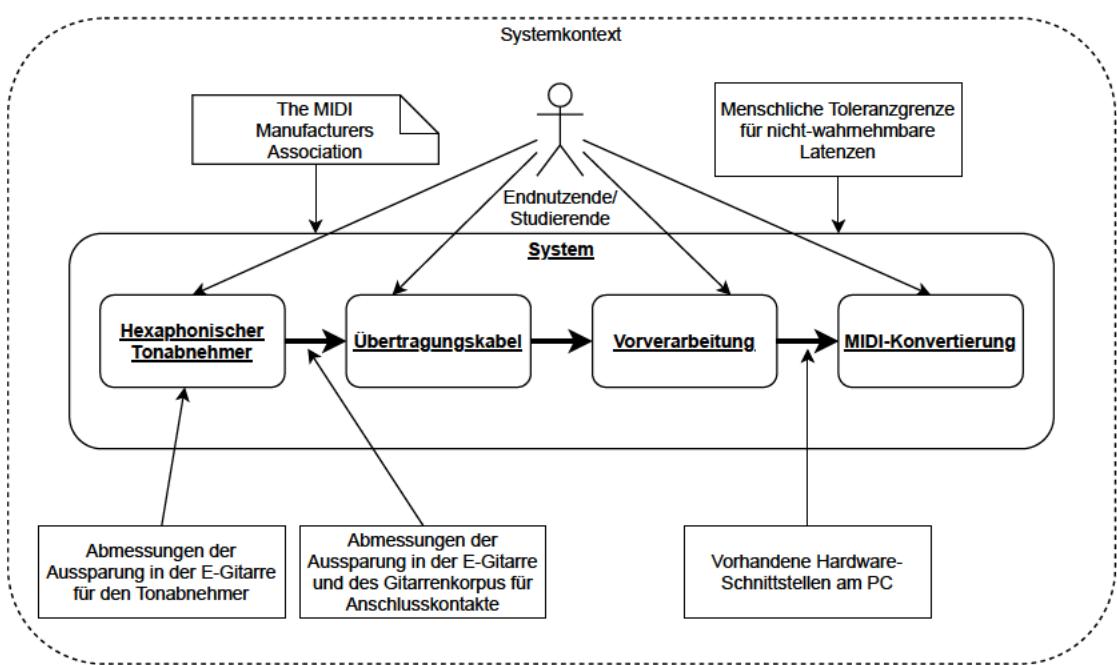


Abbildung 4.2: Systemkontext

Systemgrenzen und -schnittstellen: Für den hexaphonischen Tonabnehmer und das Instrumentenkabel bilden die Abmessungen der jeweiligen Aussparungen in der E-Gitarre und die Abmessungen des Gitarrenkorpus die Systemgrenzen und -schnittstellen in Bezug auf Montage und Interaktion mit der E-Gitarre. Beim Übergang zwischen der Einheit für die Vorverarbeitung und der MIDI-Konvertierung bilden die vorhandenen Hardware-Schnittstellen am PC die Systembegrenzung. Im Kontext von Live-Verarbeitung der Daten bildet die menschliche Toleranzgrenze für nicht-wahrnehmbare Latenzen eine weitere Systemeinschränkung. Für die zukünftige Bedienung durch Studierende gibt es keine eindeutige Schnittstelle zur Interaktion, da sie grundsätzlich an jeder Komponente direkt Änderungen oder Erweiterungen vornehmen können. Endnutzende interagieren ähnlich wie Studierende mit jeder Komponente des Systems, nur dass sie keine Intention zur Erweiterung haben.

4.3 Interessengruppen

Bei der Entwicklung des Systems müssen verschiedene Interessengruppen und ihre Anforderungen an das System beachtet werden. Diese werden im Folgenden analysiert. Ein

Teil der Interessengruppen ist im Systemkontext enthalten, der Rest setzt sich aus anderen Personengruppen zusammen, die ein anderweitiges Interesse am System haben.

Aus dem Systemkontext leiten sich die Interessengruppen Studierende, die zukünftig am System arbeiten, Endnutzende, die zukünftig das System benutzen wollen, und die MMA ab.

Studierende: Zukünftige Studierende haben ein Interesse daran, dass das System eine vollständige und funktionsfähige Verarbeitungskette bietet und sich einfach benutzen lässt. Im Kontext von zukünftigen Arbeiten und Projekten haben sie ein Interesse daran, dass das System leicht erweiterbar ist und vollständig dokumentiert ist, damit eine nötige Einarbeitung so kurz wie möglich ausfällt.

Endnutzende: Zukünftige Endnutzende haben ein Interesse daran, dass das System eine vollständige und funktionsfähige Verarbeitungskette bietet und sich auch durch nicht technisch versierte Personen einfach benutzen lässt. Außerdem haben sie ein Interesse daran, dass das MIDI-Format korrekt umgesetzt wird, um eine Kompatibilität mit anderen MIDI-Anwendungen zu gewährleisten.

The MIDI Manufacturers Association: Die MMA wird als Interessengruppe aufgeführt, da die von ihr betreuten Vorgaben zum Format MIDI 1.0 Anforderungen an das System stellen.

Die Interessengruppen, die nicht direkt im Systemkontext enthalten sind, sind der betreuende Erstprüfer und die Erstellerin dieser Arbeit.

Betreuerer Erstprüfer: Der betreuende Erstprüfer hat ein Interesse daran, dass das System eine vollständige und funktionsfähige Verarbeitungskette bietet. Für zukünftige Arbeiten und Projekte ist es wichtig, dass das System modular aufgebaut ist, sodass einzelne Komponenten erweitert oder sogar ausgetauscht werden können. Außerdem hat er ein Interesse an wirtschaftlicher Materialbeschaffung, an dem erlangten Wissen und der vollständigen Dokumentation des Systems. Zudem hat er ein Interesse daran, das System auch zu Demonstrationszwecken nutzen zu können.

Erstellerin dieser Arbeit: Die Erstellerin dieser Arbeit hat ein Interesse daran neue fachliche Fertigkeiten zu erlangen und dass das System eine stabile Grundfunktionalität aufweist. Außerdem hat sie ein Interesse daran, dass zukünftige Arbeiten und Projekte das System ohne lange Einarbeitungszeit nutzen können.

4.4 Anwendungsfälle

Nach der Beschreibung des Systemkontextes erfolgt die Analyse der Anwendungsfälle. Der hexaphonische Tonabnehmer und das Instrumentenkabel haben jeweils nur einen Anwendungsfall: Saitenschwingungen in elektrische Tonsignale konvertieren und diese Tonsignale übertragen. Die Anwendungsfälle der Vorverarbeitung und der Software im PC sind in den Abbildungen 4.3 und 4.4 dargestellt und werden im Folgenden näher beschrieben.

4.4.1 Vorverarbeitung

Die im Rahmen dieser Arbeit wichtigsten Anwendungsfälle der Vorverarbeitung sind in Abbildung 4.3 weiß dargestellt. Optionale beziehungsweise erst zukünftig relevante Anwendungsfälle sind grau dargestellt.

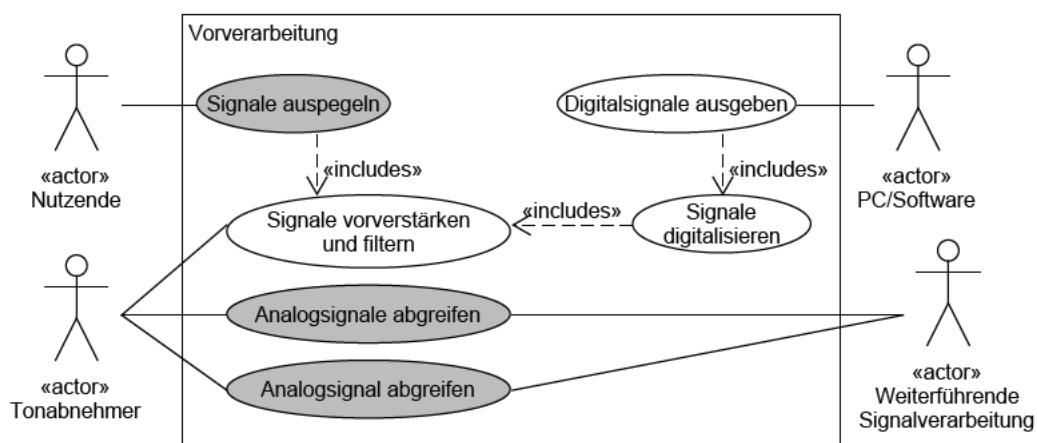


Abbildung 4.3: Anwendungsfälle der Vorverarbeitung

Signale vorverstärken und filtern Die elektrischen Signale, die von dem Tonabnehmer erzeugt werden, haben einen Instrumenten-Pegel und müssen auf den Line-Pegel verstärkt werden, bevor eine Verarbeitung durch Audio-Editoren oder DAW-Software stattfinden kann. Zusätzlich müssen sie mit mindestens 2350 Hz abgetastet werden, um das Nyquist-Shannon-Abtasttheorem für den höchsten auf der Gitarre spielbaren Ton einzuhalten [44]. Audiosignale werden üblicherweise mit mindestens 44,1 kHz abgetastet, da der menschliche Hörbereich zwischen 20 Hz und 20 kHz liegt [45].

Signale digitalisieren Damit die elektrischen Signale in der Software bearbeitet werden können, müssen sie digitalisiert werden.

Digitalsignale ausgeben Die digitalisierten Signale müssen an den Computer zur Bearbeitung übertragen werden.

Signale auspegeln Da die Vorverstärkung alle elektrischen Signale um den gleichen Faktor auf den Line-Pegel verstärkt, können Nutzende nachträglich die Einzelsignale manuell durch zusätzliche Verstärkung und Dämpfung auspegeln.

Analogsignale separat abgreifen Zukünftige Projekte und Arbeiten können vor der Vorverstärkung die Analogsignale separat für eine weiterführende Signalverarbeitung abgreifen.

Analogsignale zusammengefasst abgreifen Die sechs Analogsignale können auch zu einem einzigen Signal zusammengefasst und anschließend abgegriffen werden, dies entspricht dem Signal eines herkömmlichen Tonabnehmers.

4.4.2 MIDI-Konvertierung

Die im Rahmen dieser Arbeit wichtigsten Anwendungsfälle der MIDI-Konvertierung sind in Abbildung 4.4 weiß dargestellt. Optionale beziehungsweise erst zukünftig relevante Anwendungsfälle sind grau dargestellt.

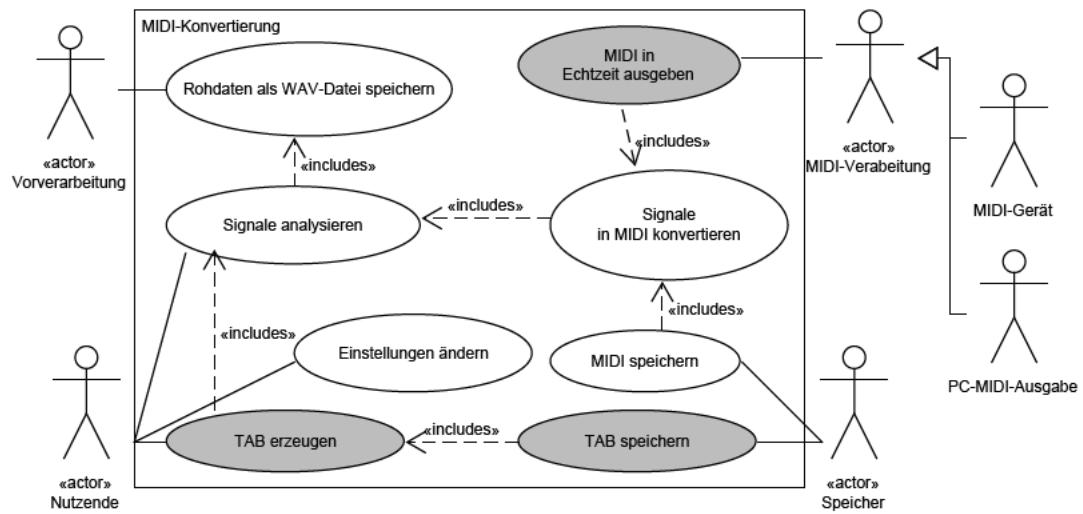


Abbildung 4.4: Anwendungsfälle der MIDI-Konvertierung

Rohdaten als WAV-Datei speichern Die ankommenden Rohdaten der Vorverarbeitung werden als Waveform Audio File Format (WAV)-Datei gespeichert. Viele Programmiersprachen bieten Bibliotheken an, mit denen WAV-Dateien analysiert und bearbeitet werden können.

Signale analysieren Die gespeicherten Rohdaten müssen in Bezug auf Frequenz, Amplitude und Tonlänge analysiert werden, um feststellen zu können, welcher Ton auf welcher Saite angeschlagen wurde.

Einstellungen ändern Für die Signalanalyse können Nutzende Einstellungen bezüglich des erwarteten Frequenzbereiches der Signale, der Analysefenstergröße und der Analyseschrittweite vornehmen.

Signale in MIDI konvertieren Analysierte Signale können in das MIDI-Format übersetzt werden, um von MIDI-fähigen Geräten oder Programmen verwendet werden zu können.

Im MIDI-Format abspeichern MIDI-Daten können auf dem PC abgespeichert werden.

MIDI in Echtzeit ausgeben MIDI-Daten können in Echtzeit ausgegeben werden, um zum Beispiel die gespielten Töne nach der Verarbeitungskette auch direkt hören zu können oder durch andere MIDI-Geräte weiterzuverwenden.

Tabulatur erzeugen Anhand der Signalanalyse kann eine Tabulatur erzeugt werden, die die gespielten Noten aufzeichnet. Dazu müssen Nutzende Informationen zur Taktart und der Spielgeschwindigkeit angeben.

Tabulatur abspeichern Die erzeugte Tabulatur kann auf dem PC als MusicXML-Datei abgespeichert werden.

4.5 Anforderungen

Nach der Analyse des Systems, der Interessengruppen und der Anwendungsfälle werden die daraus resultierenden Anforderungen abgeleitet. Sie werden in funktionale und nicht-funktionale Anforderungen unterteilt. Es werden sämtliche Anforderungen aufgeführt, die sich zum jetzigen Stand der Arbeit an das System ergeben, auch wenn nicht alle im Rahmen dieser Arbeit umgesetzt werden können.

Die Anforderungen erhalten eine Priorisierung von 0 bis 2, wobei 2 die höchste Priorität bedeutet. Anforderungen mit der Priorität 2 müssen im Rahmen dieser Arbeit mindestens teilweise umgesetzt werden. Anforderungen mit der Priorität 1 sind zusätzliche Anforderungen, die im Rahmen dieser Arbeit beachtet werden können. Anforderungen mit der Priorität 0 haben im Rahmen dieser Arbeit keine Relevanz, da sie für zukünftige Anwendungen gedacht sind.

4.5.1 Hexaphonischer Tonabnehmer

Aufgrund der Zielsetzung dieser Arbeit muss der Tonabnehmer eine hexaphonische Bauart für sechs Saiten aufweisen und in der Lage sein, die Schwingung jeder Saite einer E-Gitarre in ein eigenes elektrisches Signal gleicher Frequenz und gleichen Amplitudenver-

Tabelle 4.1: Funktionale Anforderungen an den Tonabnehmer

Nr.	Priorität	Anforderung
T-1	2	Der Tonabnehmer muss eine hexaphonische Bauart für sechs Saiten haben.
T-2	2	Der hexaphonische Tonabnehmer muss Saitenschwingungen in einzelne elektrische Signale mit gleicher Frequenz und Amplituderverlauf umwandeln können.

Tabelle 4.2: Nicht-funktionale Anforderungen an den Tonabnehmer

Nr.	Priorität	Anforderung
T-3	2	Der hexaphonische Tonabnehmer muss die Gehäuseform eines Single Coil-Tonabnehmers haben.
T-4	1	Es dürfen keine irreversiblen Maßnahmen zur Platzschaffung an der E-Gitarre durchgeführt werden.
T-5	1	Die Montage beinhaltet nur wieder lösbar Kontaktwege.

laufs umzuwandeln. Diese Anforderungen müssen zwingend umgesetzt werden, denn ohne einen funktionierenden Tonabnehmer kann keine funktionierende Verarbeitungskette aufgebaut werden. Die funktionalen Anforderungen sind in Tabelle 4.1 zusammengefasst. Bei den nicht-funktionalen Anforderungen gilt, dass der hexaphonische Tonabnehmer die Gehäusemaße eines Single Coil-Tonabnehmers haben soll, damit er in Aussparungen für Single Coil- und Humbucker-Tonabnehmer passt. Dadurch soll verhindert werden, dass an der E-Gitarre Platz schaffende Maßnahmen wie Fräsen oder Bohren durchgeführt werden müssen. Außerdem sollen für die Montage nur wieder lösbar Kontaktwege genutzt werden, um den Tonabnehmer austauschen zu können. Die nicht-funktionalen Anforderungen an den Tonabnehmer sind in Tabelle 4.2 zusammengefasst.

4.5.2 Instrumentenkabel

Das Instrumentenkabel hat die funktionale Anforderung, sechs Datensignale und sechs Masseverbindungen zwischen E-Gitarre und Vorverarbeitung übertragen zu können, siehe Tabelle 4.3. Außerdem hat es die nicht-funktionale Anforderung, dass die Anschlusskontakte auf Seiten der E-Gitarre so montiert sein sollen, dass das Spielen und Bedienen der

Tabelle 4.3: Funktionale Anforderungen an das Instrumentenkabel

Nr.	Priorität	Anforderung
K-1	2	Das Instrumentenkabel muss sechs elektrische Signale und sechs Masseverbindungen von der E-Gitarre zur Vorverarbeitung übertragen können.

Gitarre nicht beeinträchtigt werden. Zusätzlich sollen die Kontaktwege beschriftet sein, um sie den jeweiligen Tonsignalen zuordnen zu können. Die nicht-funktionalen Anforderungen an das Instrumentenkabel sind in Tabelle 4.4 zusammengefasst.

Tabelle 4.4: Nicht-funktionale Anforderungen an das Instrumentenkabel

Nr.	Priorität	Anforderung
K-2	2	Die Anschlusskontakte müssen so montiert sein, dass sie Spiel und Bedienung der E-Gitarre nicht beeinflussen.
K-3	2	Es muss erkennbar sein, welcher Signalkontakt welches Tonsignal überträgt.

4.5.3 Vorverarbeitung

Die Vorverarbeitung muss funktional in der Lage sein, die elektrischen Signale des hexaphonischen Tonabnehmers auf Line-Pegel zu verstärken und mit mindestens 2350 Hz abzutasten. Außerdem muss sie die verstärkten Signale digitalisieren und an einen Computer übertragen. Optional bietet sie die Möglichkeit, die vorverstärkten Signale nachträglich auszupegeln. Für das Auspegeln wird ein visuelles Feedback benötigt. Die elektrischen Signale können auch ohne Vorverstärkung einzeln parallel abgegriffen werden oder zusammengefasst parallel abgegriffen werden, um das Ausgangssignal eines herkömmlichen Tonabnehmers zu simulieren. Die funktionalen Anforderungen an die Vorverarbeitung sind in Tabelle 4.5 zusammengefasst.

4.5.4 MIDI-Konvertierung

Die MIDI-Konvertierung muss funktional in der Lage sein, die digitalisierten Rohdaten zunächst als WAV-Datei zu speichern. Anschließend muss sie die Daten in Bezug

Tabelle 4.5: Funktionale Anforderungen an die Vorverarbeitung

Nr.	Priorität	Anforderung
V-1	2	Die Vorverarbeitung muss die elektrischen Signale auf Line-Pegel verstärken.
V-2	2	Die Vorverarbeitung muss die elektrischen Signale mit mindestens 2350 Hz abtasten.
V-3	2	Die Vorverarbeitung muss die analogen Signale digitalisieren.
V-4	2	Die Vorverarbeitung muss die digitalen Signale an einen Computer übertragen.
V-5	0	Nutzende können die elektrischen Signale nach der Vorverstärkung manuell auspegeln.
V-6	0	Die Vorverarbeitung muss ein visuelles Feedback zum Auspegeln bieten.
V-7	1	Die Vorverarbeitung bietet die Möglichkeit, die elektrischen Signale einzeln parallel abzugreifen.
V-8	0	Die Vorverarbeitung bietet die Möglichkeit, die elektrischen Signale zusammengefasst parallel abzugreifen.

auf Tonfrequenz, Amplitude und Tonlänge analysieren. Außerdem muss sie den konkreten Tonnamen ermitteln können. Mit diesen Informationen muss sie die Signale in das MIDI-Format konvertieren und diese MIDI-Daten auf dem Computer abspeichern können. Um Einstellungen für die Analyse ändern zu können, müssen Nutzende über eine Bedienoberfläche mit der Software interagieren können. Optional kann die Software die konvertierten MIDI-Daten live für weitere Verarbeitung oder zum Anhören ausgeben. Parallel zur MIDI-Konvertierung kann die Software optional eine Tabulatur aus den analysierten Signalen aufzeichnen und als MusicXML-Datei abspeichern. Die funktionalen Anforderungen an die Software sind in Tabelle 4.6 zusammengefasst. Bei den nicht-funktionalen Anforderungen gilt, dass die Software für die MIDI-Konvertierung und die Tabulatur-Aufzeichnung aus zwei separaten Software-Modulen besteht. Außerdem muss sie auf einem Windows-Betriebssystem ab Version 10 lauffähig sein. Bei der MIDI-Konvertierung müssen die technischen Spezifikationen der MMA beachtet werden. Die nicht-funktionalen Anforderungen an die Software sind in Tabelle 4.7 zusammengefasst.

Tabelle 4.6: Funktionale Anforderungen an die MIDI-Konvertierung

Nr.	Priorität	Anforderung
S-1	1	Die Software muss die Möglichkeit zur Speicherung der Rohdaten als WAV-Datei bieten.
S-2	2	Die Software muss anhand der elektrischen Signale erkennen können, welche Töne wie laut und wie lange auf welcher Saite gespielt wurden.
S-3	2	Nutzende können den erwarteten Frequenzbereich, die Analysefenstergröße und die Analyseschrittweite ändern.
S-4	1	Über die Bedienoberfläche können Nutzende einstellen, ob eine Tabulatur erzeugt und/oder eine MIDI-Konvertierung durchgeführt wird.
S-5	1	Über die Bedienoberfläche können Nutzende einstellen, ob MIDI-Daten gespeichert und/oder live ausgegeben werden.
S-6	2	Die Software kann die analysierten Signale in das MIDI-Format konvertieren.
S-7	2	Die Software muss die MIDI-Daten im Format .mid abspeichern.
S-8	0	Die Software kann die MIDI-Daten sofort nach Erzeugung ausgeben.
S-9	0	Die Software kann eine Tabulatur erzeugen.
S-10	0	Die Software kann die erzeugte Tabulatur im Format .xml abspeichern.

4.5.5 Allgemein

Anschließend gibt es noch nicht-funktionale Anforderungen an das gesamte System. Der Aufbau und die Technik müssen nachvollziehbar und vollständig dokumentiert werden. Außerdem darf die gesamte Verarbeitungskette von Tonanschlag bis Live-MIDI-Ausgabe eine Latenz von 16 ms nicht überschreiten, siehe Tabelle 4.8.

Tabelle 4.7: Nicht-funktionale Anforderungen an die MIDI-Konvertierung

Nr.	Priorität	Anforderung
S-11	1	Die Software für die MIDI-Konvertierung und die Tabulatur-Erzeugung besteht aus zwei separaten Software-Modulen.
S-12	2	Die Software muss auf einem Windows-Betriebssystem ab Version 10 lauffähig sein.
S-13	2	Bei der Verwendung des MIDI-Formats müssen die technischen Spezifikationen der MMA für MIDI 1.0 beachtet werden.
S-14	0	Bei der Verwendung des MusicXML-Formats müssen die technischen Spezifikationen für MusicXML Version 4.0 beachtet werden.

Tabelle 4.8: Nicht-funktionale Anforderungen an das gesamte System

Nr.	Priorität	Anforderung
A-1	2	Für jedes eingebaute Bauteil muss sein Zweck, seine Bedienung und mögliche einstellbare Parameter dokumentiert sein.
A-2	1	Die Verarbeitungskette für die Live-MIDI-Ausgabe darf eine Latenz von 16 ms nicht überschreiten.

5 Gesamtkonzept

In diesem Kapitel wird das Gesamtkonzept und die Vorgehensweise der Arbeit erstellt, bevor in den folgenden Kapiteln auf die einzelnen Komponenten konkret eingegangen wird.

5.1 Schnittstellen

Die Verarbeitungskette beinhaltet folgende Schnittstellen:

- E-Gitarre ↔ hexaphonischer Tonabnehmer
- Hexaphonischer Tonabnehmer ↔ Gitarrenkabel
- Gitarrenkabel ↔ Vorverarbeitung
- Vorverarbeitung ↔ Digitale Signalanalyse und -verarbeitung

Die Schnittstelle zwischen E-Gitarre und hexaphonischen Tonabnehmern besteht aus den Montagemöglichkeiten und dem zur Verfügung stehenden Platz für einen Tonabnehmer innerhalb der E-Gitarre. Nach den Anforderungen T-3 und T-5 muss Platz für einen Single Coil-Tonabnehmer vorhanden sein und der Tonabnehmer muss wieder lösbar montiert werden, wie es mit Schraub- und Steckverbindungen möglich ist.

Die Schnittstellen zum Gitarrenkabel sind funktional auf sechs Datenverbindungen und sechs Masseverbindungen ausgelegt. Der Aufbau der mechanischen Verbindung hängt auf der Seite der Vorverarbeitung davon ab, ob diese einen bestimmten mechanischen Anschluss vorgibt oder nicht. Auf der Seite der E-Gitarre muss die mechanische Verbindung laut Anforderung K-2 so aufgebaut sein, dass sie Spiel und sonstige Bedienung der E-Gitarre nicht beeinflusst. Dementsprechend sollten Anschlusskontakte möglichst klein ausfallen und per Schraub- oder Klebekontakt montierbar sein.

Die Schnittstelle zwischen Vorverarbeitung und Computer respektive der Analyse-Software

beinhaltet die physische Schnittstelle in Form von Anschlüssen und die Software-Schnittstelle in Form eines Übertragungsprotokolls. Hierbei hängt die Entwicklungsfreiheit wieder davon ab, ob die Vorverstärkung eine Schnittstelle zur Verfügung stellt oder nicht.

5.2 Vorgehensweise

Nach der Beschreibung der Schnittstellen erfolgt die Erläuterung der Vorgehensweise.

5.2.1 Beschaffung der E-Gitarre

Als Erstes wird eine E-Gitarre mit Aussparungen für Tonabnehmer benötigt. Da im Rahmen dieser Arbeit die subjektiv wahrgenommene Klangqualität des Tonabnehmers respektive der E-Gitarre selbst nicht betrachtet wird, erfolgt die Auswahl der E-Gitarre lediglich nach Kosten-, Montage- und Lieferfaktoren. Um so viel Platz wie möglich für die Montage eines Tonabnehmers zu haben, werden E-Gitarren mit Humbucker-Tonabnehmern bevorzugt, da sie Platz für zwei Single Coil-Spulen beziehungsweise für einen Single Coil-Tonabnehmer und zusätzliche Elektronik bieten. Außerdem wird darauf geachtet, dass das Elektronikfach der E-Gitarre möglichst groß ausfällt und von hinten zu erreichen ist. So kann zusätzliche Elektronik auch dort untergebracht werden und lässt sich leicht erreichen, ohne die Saiten oder sonstige Bauteile auf der Vorderseite der E-Gitarre abbauen zu müssen. Von Sets zur Selbstmontage wird abgesehen, da deren Korpusse für gewöhnlich keine größeren Aussparungen für Tonabnehmer und Elektronik bieten, sondern nur Mehraufwand in der Montage bedeuten. In Tabelle 5.1 werden die vier günstigsten E-Gitarren mit mindestens einem Humbucker-Tonabnehmer aus dem Sortiment eines Musikhauses verglichen¹. Die Einträge H und S in der Zeile „Tonabnehmer“ stehen für Humbucker und Single-Coil.

- Gitarre 1: Harley Benton ST-20HSS SB Standard Series²
- Gitarre 2: Harley Benton DC-200 BK Student Series³
- Gitarre 3: Harley Benton TE-20HH SBK Standard Series⁴

¹Stand der Recherche: Februar 2024

²https://www.thomann.de/de/harley_benton_st_20hss_sb_standard_series.htm

³https://www.thomann.de/de/harley_benton_dc_200_bk_student_series.htm

⁴https://www.thomann.de/de/harley_benton_te_20hh_sbk_standard_series.htm

Tabelle 5.1: Vergleich geeigneter E-Gitarren

	Gitarre 1	Gitarre 2	Gitarre 3	Gitarre 4
Tonabnehmer	HSS	HH	HH	HH
Elektronikfach	hinten	hinten	vorne	hinten
Preis	99 €	99 €	99 €	109 €
Lieferzeit	8-10 Wochen	lieferbar	7-9 Wochen	lieferbar

- Gitarre 4: Harley Benton SC-200 BK Student Series⁵

Von diesen vier E-Gitarren werden die Nr. 2 und die Nr. 4 näher betrachtet, da die Nr. 1 eine zu lange Lieferzeit und die Nr. 3 kein von hinten erreichbares Elektronikfach hat. Obwohl die Nr. 2 günstiger als die Nr. 4 ist, hat sie den Produktbildern nach zu urteilen ein kleineres Elektronikfach, sodass die Nr. 4 für die Arbeit beschafft wird.

5.2.2 Konstruktion eines hexaphonischen Tonabnehmers

Anschließend werden mehrere hexaphonische Tonabnehmer mit unterschiedlichen Wicklungszahlen gewickelt, um eine Auswahl verschiedener Aufbauten zu haben. Unabhängig von der Wicklung müssen die Tonabnehmer auf jeden Fall ein Single Coil-Format aufweisen und in die Aussparungen in der E-Gitarre passen können. Die Konzipierung und Umsetzung werden in Kapitel 6 aufgeführt.

5.2.3 Beschaffung der Vorverarbeitung

Wie in Kapitel 4.4.1 beschrieben, muss die Vorverarbeitung die analogen Signale mindestens verstärken und digitalisieren können. Das kann man entweder selbst konstruieren oder eine fertige Marktlösung verwenden. Da im Rahmen dieser Arbeit der konstruktive Schwerpunkt auf dem Aufbau eines hexaphonischen Tonabnehmers und der Programmierung der MIDI-Konvertierung liegt, wird aus Zeit- und Aufwandsgründen eine fertige Marktlösung bevorzugt. Das Audiointerface muss mindestens sechs Instrumenteneingänge besitzen, damit alle erzeugten Signale des hexaphonischen Tonabnehmers gleichzeitig bearbeitet werden können. Außerdem muss das Interface die Anforderungen V-1 bis V-8 erfüllen.

⁵https://www.thomann.de/de/harley_benton_sc_200_bk_student_series.htm

Tabelle 5.2: Vergleich geeigneter Audiostreaming-Interfaces

	Scarlett 18i20	UMC1820	8 Pre USB
Instrumenteneingänge	2	8	8
Abtastrate	192 kHz	96 kHz	96 Hz
Auflösung	24 bit	24 bit	24 bit
Auspeichern mit Feedback	ja	ja	ja
Eingänge einzeln parallel abgreifen	ja	mit Hardware-Latenz	ja
Eingänge zusammengefasst parallel abgreifen	ja	ja	ja
Schnittstelle zum PC	USB 2.0	USB 2.0	USB 2.0
Preis	-	262 €	739 €
Verfügbarkeit	-	lieferbar	lieferbar

Aus dem Labor für Signalverarbeitung der HAW Hamburg steht ein Audiostreaming-Interface des Typs „Scarlett 18i20“⁶ für interne Arbeiten zur Verfügung. Die Marktrecherche hat gezeigt, dass die meisten Audiostreaming-Interfaces für den nicht-professionellen Bereich nur über zwei Instrumenteneingänge verfügen. Das liegt daran, dass eine typische Band-Zusammensetzung aus einer E-Gitarre, einer Bass-Gitarre, mehreren Mikrofonen und sonstigen Instrumenten mit Line-Pegel besteht. Sollte man mehr als zwei Gitarren haben, verwendet man Mikrofone, die nahe an den Instrumenten aufgestellt werden. Beim hexaphonischen Tonabnehmer bedeutet dies, dass man nur zwei Saiten gleichzeitig aufnehmen kann. Das Testen von Akkorden über alle Saiten wird damit sehr schwierig, da die Saiten nur nacheinander aufgenommen werden können und der Akkord für die Vergleichbarkeit mindestens dreimal auf die gleiche Weise gespielt werden muss, was manuell kaum möglich ist. In Tabelle 5.2 werden die zwei Audiostreaming-Interfaces^{7,8} mit mindestens sechs Instrumenteneingängen aus dem Sortiment eines Musikhauses und das Interface aus dem Labor verglichen⁹. Von den Interfaces wird aufgrund der technischen Daten und der Preises das Interface „UMC1820“ bevorzugt, dessen Anschaffungspreis aber außerhalb des Projekt-budgets liegt. Freundlicherweise hat uns die Firma „No.1 Guitar Center GmbH“¹⁰ das benötigte Interface als Dauerleihgabe zur Verfügung gestellt. Für diese wertvolle Unter-

⁶https://www.thomann.de/de/focusrite_scarlett_18i20_3rd_gen.htm

⁷https://www.thomann.de/de/behringer_umc1820.htm

⁸https://www.thomann.de/de/motu_8_pre_usb.htm

⁹Stand der Recherche: April 2024

¹⁰<https://www.n1-guitars.de/>

stützung, ohne die diese Arbeit im Bereich der Gitarren-Technologie in der derzeitigen Form nicht möglich gewesen wäre, bedanken wir uns daher ganz ausdrücklich.

5.2.4 Beschaffung des Gitarrenkabels

Nachdem das Audiointerface beschafft wurde, wird das Gitarrenkabel zwischen hexaphonischen Tonabnehmer und Audiointerface benötigt. Auf der Seite des Audiointerfaces muss das Gitarrenkabel sechs 1/4 in-Klinkenstecker zur Verfügung stellen, auf der Seite des Tonabnehmers und der Gitarre ist die Schnittstelle frei konzipierbar. Die Konzipierung und die Umsetzung für das Gitarrenkabel und die Anschlusskontakte erfolgen in Kapitel 7.

5.2.5 Zugriff und Visualisierung der digitalisierten Daten

Nach der Beschaffung der Hardware erfolgt der softwareseitige Zugriff auf die Daten des Audiointerfaces. Da das beschaffte Audiointerface selbst keine dazugehörige DAW-Software besitzt oder auf eine beschränkt ist, kann die DAW-Software frei gewählt werden. Dabei sind die Anforderungen V-7 und A-2 zu beachten. Außerdem soll sie, wenn möglich, kostenlos nutzbar sein. Um alle sechs Eingangskanäle mit geringer Latenz auslesen zu können, wird für die meisten Computer-Soundkarten ein Audio Stream Input/Output (ASIO)-Treiber benötigt [33]. Ohne ihn können sonst nur zwei Audiokanäle genutzt werden. Die Gerätetreiber für das genutzte Audiointerface beinhalten einen gerätekompatiblen ASIO-Treiber, sodass bei der Auswahl der DAW-Software auf eine entsprechende Kompatibilität geachtet werden muss. Daher wird der Audio-Editor „Wavosaur“¹¹ genutzt, der frei verfügbar ist und ASIO-Treiber unterstützt.

5.2.6 Auswahl eines PDA zur Tonanalyse

Nachdem die aufgenommenen Daten als WAV-Datei vorliegen, gilt es die gespielten Töne zu erkennen. Von den in Kapitel 3.5 vorgestellten PDA wird der YIN-Algorithmus verwendet, da eine fertige Implementierung verfügbar ist, die zugrunde liegenden Ansätze und Formeln leicht nachvollziehbar und anpassbarer als bei maschinellen Lernen und neuronalen Netzen sind und die Genauigkeit bei der Analyse hoch ist. Dabei wird in

¹¹<https://www.wavosaur.com/>

Kauf genommen, dass YIN nicht echtzeitfähig arbeiten kann. Die YIN-Implementierung liegt zwar in MATLAB vor, enthält aber keine Funktionen aus speziellen Toolboxen oder Bibliotheken, sodass sie mit geringem Aufwand in der frei verfügbaren Programmierumgebung GNU Octave eingesetzt werden kann.

5.2.7 Tonanalyse

Als Nächstes folgt die Vorgehensweise bei der Tonanalyse. Der generelle Aufbau der Analyse-Software, die konkrete Einbindung des YIN-Algorithmus und die MIDI-Konvertierung werden in Kapitel 8 aufgeführt.

5.2.8 Verwendung von Audiodatenbanken

Unter [55] kann eine Datenbank für Gitarrenmusik heruntergeladen werden. Die einzelnen Aufnahmen beinhalten sowohl monophone als auch polyphone Musik verschiedener Genres. Zu jeder Aufnahme gibt es auch eine MIDI-Datei mit Anmerkungen zu Tonhöhen, Tonlängen, Taktschlägen und dem Liedtempo. Es werden sowohl ein klassischer Tonabnehmer als auch ein hexaphonisches Tonabnehmer eingesetzt, sodass jede Aufnahme einkanalig als auch sechskanalig vorliegt. Da aber ein hexaphonisches Tonabnehmer für Schalllöcher von Akustikgitarren verwendet wurde, ist der Saitenabstand deutlich höher als bei Tonabnehmern für E-Gitarren, sodass viel Interferenz zwischen benachbarten Saiten aufgenommen wird. Deswegen wird vom Einsatz der Datenbank abgesehen und es werden eigene Samples eingespielt.

Tonanalyse einer Saite mit statischen Ton

Zuerst wird auf einer Saite ein Ton erzeugt und analysiert. Das wird mit verschiedenen Tönen auf der Saite wiederholt, sodass für jeden Ton eine einzelne WAV-Datei vorliegt. Damit wird die Grundperformanz des YIN-Programms getestet.

Tonanalyse einer Saite mit mehreren Tönen nacheinander

Anschließend werden mehrere Töne inklusive Pausen auf einer Saite gespielt und als eine WAV-Datei abgespeichert. Damit wird die zeitliche Auflösung und der Umgang mit Pausen getestet.

Tonanalyse aller Saiten mit einem Ton

Als Nächstes wird ein Ton auf einer Saite gespielt, aber die Signale aller Saiten aufgenommen, sodass für einen gespielten Ton sechs Dateien mit jeweils einem Ton vorliegen. Damit wird die Interferenz der einzelnen Tonabnehmer analysiert.

Tonanalyse aller Saiten mit mehreren Tönen auf verschiedenen Saiten

Dann werden mehrere Töne nacheinander auf verschiedenen Saiten gespielt und alle Saiten aufgenommen. Das dient zum Testen, ob die Software korrekt unterscheiden kann, auf welcher Saite welcher Ton gespielt wurde.

Tonanalyse von Akkorden

Zuletzt werden mehrere Töne auf verschiedenen Saiten gleichzeitig gespielt, um zu testen, ob der YIN-Algorithmus in Kombination mit einem polyphonen Tonabnehmer in der Lage ist, polyphone Klänge zu analysieren.

5.2.9 MIDI-Konvertierung

Für die MIDI-Konvertierung wird eine Auflistung aus Tonhöhen, Tonlängen, Lautstärken und Pausenzeiten benötigt. Die Tonhöhe, Tonlänge und die Pausenzeiten können aus dem YIN-Algorithmus abgeleitet werden, die Ermittlung der Lautstärke muss noch ergänzt werden. Da am Ende der Tonanalyse im Idealfall sechs unabhängige aber zeitgleiche Ergebnisse für die sechs Saiten vorliegen, wird für die MIDI-Datei das Format 1 mit sechs Tracks gewählt. Die MIDI-Datei wird mit Funktionen zur Datei-Manipulation in GNU Octave erstellt. Zum Auslesen der erstellten Dateien wird das frei verfügbare Programm

„MidiEditor“¹² verwendet. Es zeigt die MIDI-Töne als horizontale Balken an und kann sie abspielen und auch nachträglich bearbeiten.

¹²<http://www.midieditor.org/>

6 Hexaphonischer Tonabnehmer

In diesem Kapitel werden das Konzept und die Umsetzung zum Bau eines hexaphonischen Tonabnehmers im Single Coil-Format beschrieben. Zunächst erfolgt die Festlegung der geometrischen Parameter bezüglich Gehäuseaufbau, Magnetmaterial und Spulendraht. Danach wird das Verfahren zum Spulenwickeln und zum Einbau in der Gitarre konzipiert. Anschließend erfolgt die Beschreibung der Amplitudenganganalyse. Im Unterkapitel zur Umsetzung werden die durchgeführten Schritte und ihre Ergebnisse vorgestellt. Parallel zur Erstellung dieser Arbeit hat der betreuende Erstprüfer aufgrund der gemachten Erfahrungen der Autorin dieser Arbeit parallel eine Weiterentwicklung des hexaphonischen Tonabnehmers erarbeitet. Diese wird am Ende des Kapitels vorgestellt.

6.1 Konzept

Zunächst erfolgt die Konzipierung des hexaphonischen Tonabnehmers vom Spulenwickeln bis zum Einbau in der Gitarre.

6.1.1 Aufbau des Gehäuses

Das Gehäuse des hexaphonischen Tonabnehmers wird so gestaltet, dass es eine Single Coil-Bauform aufweist und in die Aussparungen für die Humbucker-Tonabnehmer in der E-Gitarre passen. Dazu wird mit einem 3D-Drucker ein passendes Gehäuse gedruckt, das auch Trennwände für die sechs Spulen beinhaltet. Das 3D-Modell wird vom betreuenden Erstprüfer in Anlehnung an Single Coil-Tonabnehmer für Stratocaster-Gitarren erstellt und ist in Abbildung 6.1 zu sehen. Auf der Unterseite ist ein schmaler Spalt eingelassen, um die Spulendrahte aus dem Gehäuse führen zu können. Nicht dargestellt ist die dazugehörige Bodenplatte, die die gleiche Grundfläche wie das Gehäuse hat und auch sechs Öffnungen für die Spulenmagnete beinhaltet. Sie dient zur Stabilisierung und Ausrichtung der Spulenmagnete.



Abbildung 6.1: 3D-Modell eines Gehäuses für hexaphonische Tonabnehmer im Single Coil-Format

6.1.2 Auswahl der Magneten

Für die Spulenmagneten wird eine Legierung aus Aluminium, Nickel und Kobalt (AlNi-Co) verwendet, wobei die Mischung AlNiCo 5 am weit verbreitesten ist [6, 57]. Für den hexaphonischen Tonabnehmer werden passend für das Gehäuse sechs AlNiCo 5 Magnete mit einem Durchmesser von 5 mm und einer Höhe von 1,8 cm eingesetzt.

6.1.3 Auswahl des Spulendrahts

Tonabnehmer werden häufig mit Kupferlackdraht der Dicke AWG 42 (Durchmesser: 0,06334 mm) gewickelt [57]. Die Lackisolierung verhindert Kurzschlüsse zwischen den Lagen der Wicklungen. Zum Wickeln wird ein Kupferlackdraht mit einem Durchmesser von 0,08 mm inklusive Isolierung beschafft.

6.1.4 Variation der Wicklungszahl

Die genaue Anzahl der Wicklungen in Tonabnehmern ist von Hersteller zu Hersteller, von Modell zu Modell und sogar von Produktionszeitraum zu Produktionszeitraum unterschiedlich. Moderne Hochimpedanz-Tonabnehmer haben ca. 6000 Wicklungen [6], wobei Werte zwischen 5000 und 10000 Wicklungen möglich sind [57]. Niederimpedante Tonabnehmer haben zwischen 1000 und 2000 Wicklungen, sie benötigen normalerweise eine

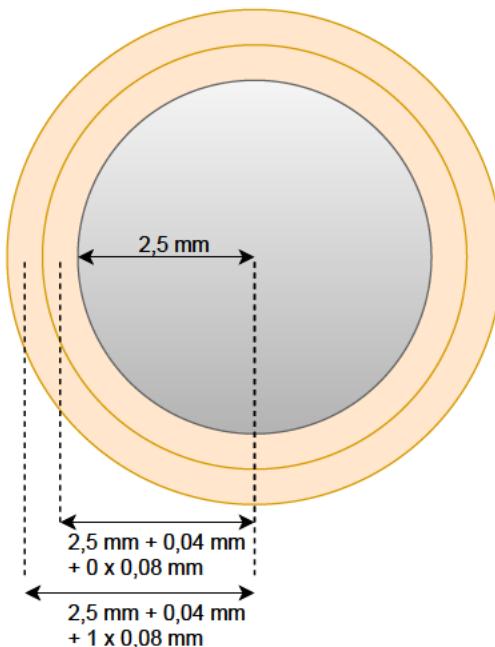


Abbildung 6.2: Berechnung der Radien der Drahtwicklungen

aktive Vorverstärkung in der E-Gitarre, um die gleiche Ausgangsspannung wie hochimpedante Tonabnehmer zu erzeugen [6].

Innerhalb des Gehäuses hat jede Spule eine Kreisfläche mit 9,8 mm Durchmesser zur Verfügung, wobei die mittleren 5 mm durch den Spulenmagnet eingenommen werden. Um den Magnet herum bleiben damit 2,4 mm für die Wicklungsringe des Spulendrahts. In der Höhe stehen 1,6 cm für die Wicklungslagen zur Verfügung. In Theorie ergeben sich für den verwendeten Kupferlackdraht mit 0,08 mm Durchmesser maximal 30 Wicklungsringe und maximal 200 Wicklungslagen, was zu einer Gesamtwicklungszahl von 6000 Wicklungen führt. Bei Handwicklung ist davon auszugehen, dass am Ende durch unvermeidbare Zwischenräume weniger Wicklungen vorliegen. Um verschiedene Wicklungszahlen vergleichen zu können, werden drei Tonabnehmer mit jeweils 30, 20 und 10 Wicklungsringen pro Spule gewickelt. Die Anzahl an Wicklungsringen muss auf jeden Fall gerade sein, damit Drahtanfang- und Ende sich beide am unteren Ende des Tonabnehmers befinden, um sie an die weiterführenden Leitungen löten zu können.

Für die Abschätzung der benötigten Länge an Kupferlackdraht pro sechs Spulen werden Kreisflächen bis zur Mitte des Drahtes eines Wicklungsringes betrachtet, siehe Abbildung 6.2, und anhand Gleichung 6.1 für die Anzahl der Wicklungsringe N_r aufsummiert.

$$l_{Cu} = 6 \cdot 200 \cdot \sum_{n=0}^{N_r-1} 2\pi \cdot (2,5 \text{ mm} + 0,04 \text{ mm} + n \cdot 0,08 \text{ mm}) \quad (6.1)$$

30 Wicklungsringe benötigen ungefähr 837 m Kupferlackdraht, 20 Wicklungsringe benötigen ungefähr 498 m und 10 Wicklungsringe benötigen ungefähr 219 m.

6.1.5 Messung des Spulenwiderstands

Um versteckten Drahtbuch in den Spulen ausschließen zu können, wird der Spulenwiderstand gemessen. Unter der Annahme, dass der spezifische Widerstand des Kupferdrahts dem Tabellenwert $0,017 \frac{\Omega \cdot \text{mm}}{\text{m}}$ [11] entspricht und der Durchmesser des Kupferdrahts zwischen 0,06 mm und 0,08 mm groß ist, lässt sich der erwartete Widerstandswert pro Spule R_{Spule} über Gleichung 6.2 abschätzen [21]:

$$R_{Spule} = \frac{\rho \cdot \frac{l_{Cu}}{6}}{A} \quad (6.2)$$

Bei 10 Wicklungsringen liegt der erwartete Widerstandswert zwischen 123,4 Ω und 219,5 Ω pro Spule:

$$R_{Spule,min} = \frac{0,017 \frac{\Omega \cdot \text{mm}^2}{\text{m}} \cdot \frac{219 \text{ m}}{6}}{\pi \cdot 0,04^2 \text{ mm}^2} = 123,4 \Omega$$

$$R_{Spule,max} = \frac{0,017 \frac{\Omega \cdot \text{mm}^2}{\text{m}} \cdot \frac{219 \text{ m}}{6}}{\pi \cdot 0,03^2 \text{ mm}^2} = 219,5 \Omega$$

$$\Rightarrow 123,4 \Omega \leq R_{Spule} \leq 219,5 \Omega$$

6.1.6 Herstellen der elektrischen Verbindung

Um die Signale des Tonabnehmers sicher übertragen zu können, werden Aderleitungen mit vorkonfektionierten Kontakten, umgangssprachlich „Jumperkabel“ genannt, verwendet, die auf einer Seite mit den Kupferlackdrähten verbunden werden müssen. Dazu wird

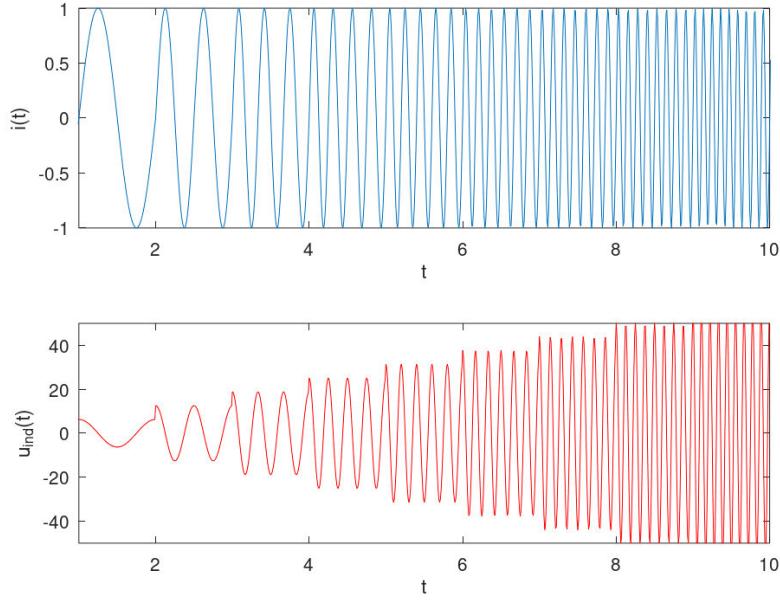
ein schmaler Streifen Lochrasterplatine an die Seitenkante der Bodenplatte des Tonabnehmergehäuses geklebt und jeweils zwölf Stiftleisten eingelötet. Die Kupferlackdrähte werden auf der Unterseite der Platine an die Stiftleisten gelötet und die Aderleitungen auf die Stiftleisten gesetzt. Um die Aderleitungen den einzelnen Tonabnehmermagneten und letztlich den Saiten der Gitarre zuordnen zu können, wird pro Saite eine Farbe für die Signal- und Masseleitung verwendet. Die Signalleitungen werden zusätzlich markiert, um sie nicht mit den Masseleitungen zu verwechseln.

6.1.7 Montage in der Gitarre

Um den hexaphonischen Tonabnehmer in der Gitarre zu montieren, wird der vorhandene Humbucker-Tonabnehmer am Halsende der Gitarre ausgebaut und dessen Ausbuchtung verwendet. Dabei wird darauf geachtet, dass die vorhandene Schaltung in der Gitarre möglichst intakt bleibt, sodass der Steg-Tonabnehmer noch weiter verwendet werden kann. Da der Hals-Tonabnehmer näher am Scheitelpunkt der Schwingung der Grundtöne der Saiten liegt, nimmt er eine höhere Amplitude auf als der Steg-Tonabnehmer, der näher am fixierten Ende der Saite liegt, siehe Abbildung 2.5 auf Seite 9.

6.1.8 Messung des Amplitudengangs

Nach der Montage des hexaphonischen Tonabnehmers wird sein Amplitudengang gemessen. Im Kontext der Klangqualität ist es wichtig, dass die Resonanzfrequenz des Tonabnehmers im typischen Bereich von 2 - 5 kHz liegt. Ohne eine Resonanzüberhöhung dieser Frequenzen würde der Klang fade und langweilig erscheinen [26]. Für einen hexaphonischen Tonabnehmer ist es aber gerade wichtig, dass die Resonanzfrequenz so hoch wie möglich liegt, damit der Frequenzbereich der Gitarre gleichmäßig übertragen wird. Während der anschließenden Signalkonvertierung wird die Amplitude des gespielten Gitarrentons analysiert und diese soll möglichst nicht durch die Übertragungscharakteristik des Tonabnehmers verfälscht werden. Dafür wird in Anlehnung an [25] eine niederimpedante Geberspule ohne Magnetkern an einen Signalgenerator angeschlossen, der eine Sinusspannung mit steigender Frequenz und proportional zur Frequenz fallender Amplitude ausgibt. Die Geberspule wird direkt auf eine Spule des hexaphonischen Tonabnehmers gelegt und deren Ausgangsspannung aufgezeichnet. Es ist wichtig, dass das Testsignal eine fallende Amplitude aufweist, da die induzierte Spannung in der Tonabnehmerspule


 Abbildung 6.3: Verlauf von $i(t)$ und $u_{ind}(t)$

nach Gleichung 2.3 von der Ableitung der magnetischen Flussdichte abhängt. Die magnetische Flussdichte einer Zylinderspule hängt nach Gleichung 6.3 wiederum von der Stromstärke $i(t)$ des Testsignals ab [11].

$$B(t) = \mu_0 \cdot \frac{N_{Test}}{l_{Test}} \cdot i(t) = \mu_0 \cdot \frac{N_{Test}}{l_{Test}} \cdot \hat{I} \cdot \sin(2\pi f \cdot t) \quad (6.3)$$

Setzt man Gleichung 6.3 in Gleichung 2.4 ein, ist in Gleichung 6.4 zu erkennen, dass die Amplitude der Ableitung eines Sinustons mit definierter Frequenz proportional zu dieser Frequenz steigt. Das heißt auch die Amplitude der induzierten Spannung in der Tonabnehmerspule steigt mit steigender Frequenz, siehe Abbildung 6.3.

$$\begin{aligned} u_{ind}(t) &= -N_{Spule} \cdot \frac{d}{dt} \int_A \mu_0 \cdot \frac{N_{Test}}{l_{Test}} \cdot \hat{I} \cdot \sin(2\pi f \cdot t) \cdot d\vec{A} \\ \Rightarrow u_{ind}(t) &= -N_{Spule} \cdot \int_A \mu_0 \cdot \frac{N_{Test}}{l_{Test}} \cdot \hat{I} \cdot 2\pi f \cdot \cos(2\pi f \cdot t) \cdot d\vec{A} \end{aligned} \quad (6.4)$$

In Gleichung 6.5 ist der Korrekturfaktor $\frac{1}{f}$ ergänzt, sodass die Amplitude der induzierten

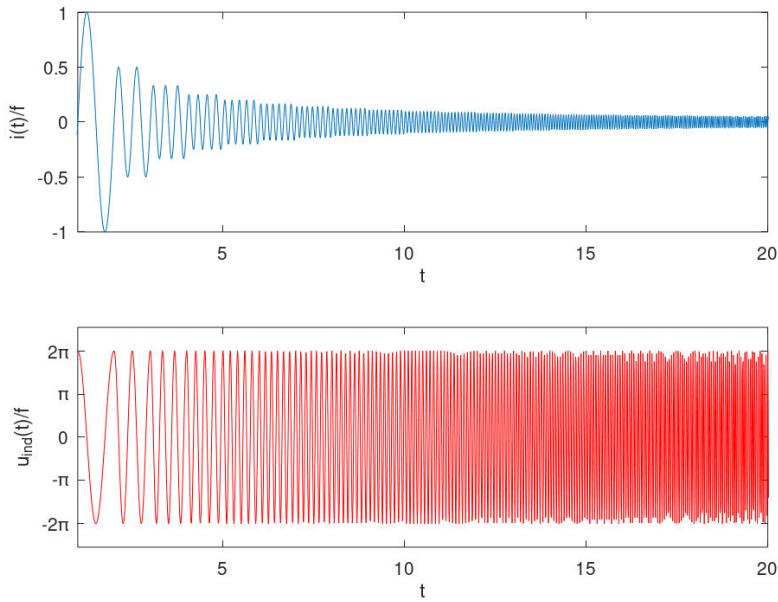


Abbildung 6.4: Verlauf von $i(t)/f$ und $u_{ind}(t)/f$

Spannung frequenzunabhängig ist, siehe Abbildung 6.4.

$$\begin{aligned}
 u_{ind}(t)/f &= -N_{Spule} \cdot \frac{d}{dt} \int_A \mu_0 \cdot \frac{N_{Test}}{l_{Test}} \cdot \frac{\hat{I}}{f} \cdot \sin(2\pi f \cdot t) \cdot d\vec{A} \\
 \Rightarrow u_{ind}(t)/f &= -N_{Spule} \cdot \int_A \mu_0 \cdot \frac{N_{Test}}{l_{Test}} \cdot \hat{I} \cdot 2\pi \cdot \cos(2\pi f \cdot t) \cdot d\vec{A}
 \end{aligned} \quad (6.5)$$

Multipliziert man nun $u_{ind}(t)$ als Eingangssignal mit der Übertragungsfunktion $H'(f)$ aus Gleichung 2.8, erhält man den roten Graph in Abbildung 6.5. Die linear ansteigende Amplitude von $u_{ind}(t)$ prägt sich dabei dem Verlauf von $H'(f)'$ auf und der Verlauf der Resonanzüberhöhung lässt sich schwerer erkennen, vor allem wenn die Resonanzgüte klein ist. Nutzt man den Korrekturfaktor, ergibt sich der gelbe Graph. Er hat den gleichen Verlauf wie $H'(f)$ im blauen Graph, ist aber um den Faktor 2π in dB verstärkt.

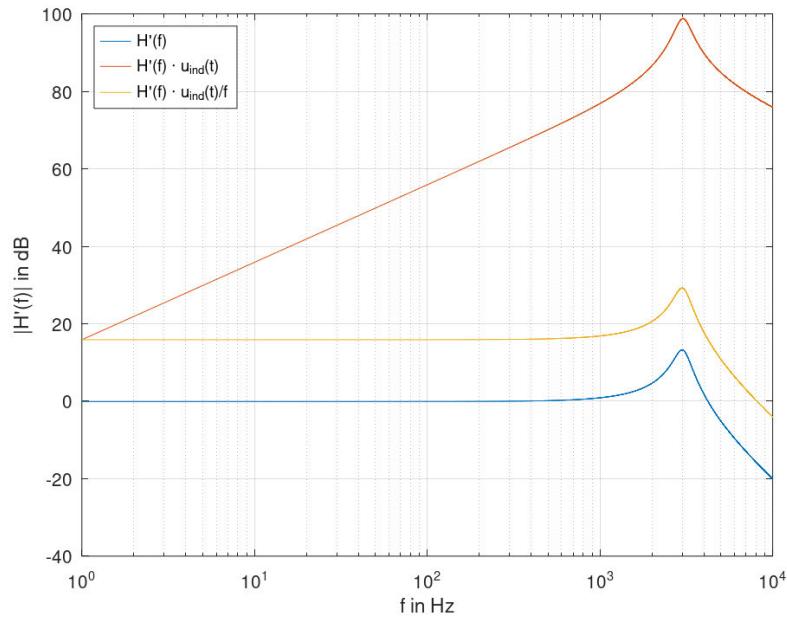


Abbildung 6.5: Verlauf von $H'(f)$, $H'(f) \cdot u_{ind}(t)$ und $H'(f) \cdot u_{ind}(t)/f$

6.2 Umsetzung

6.2.1 Gehäusedruck

Das Gehäuse samt Bodenplatte wird vom betreuenden Erstprüfer mit einem 3D-Drucker aus Polylactid (PLA) gedruckt. Der Druck weist üblicherweise einige Fertigungstoleranzen auf, sodass die Öffnungen für die Spulenmagneten nachträglich mit einer Rundfeile bearbeitet werden müssen, bis die Spulenmagneten tatsächlich passen. Außerdem werden die Unterseite des Gehäuses und die Oberseite der Bodenplatte eben geschliffen, damit sie plan aufeinander liegen.

6.2.2 Wicklung

Aufbau

Für das Wickeln der Spulen wird ein Arduino UNO mit dem Schrittmotortreiber TB6600 und einem Schrittmotor der Baugröße NEMA 17 mit einem Nennstrom von 1,5 A ver-

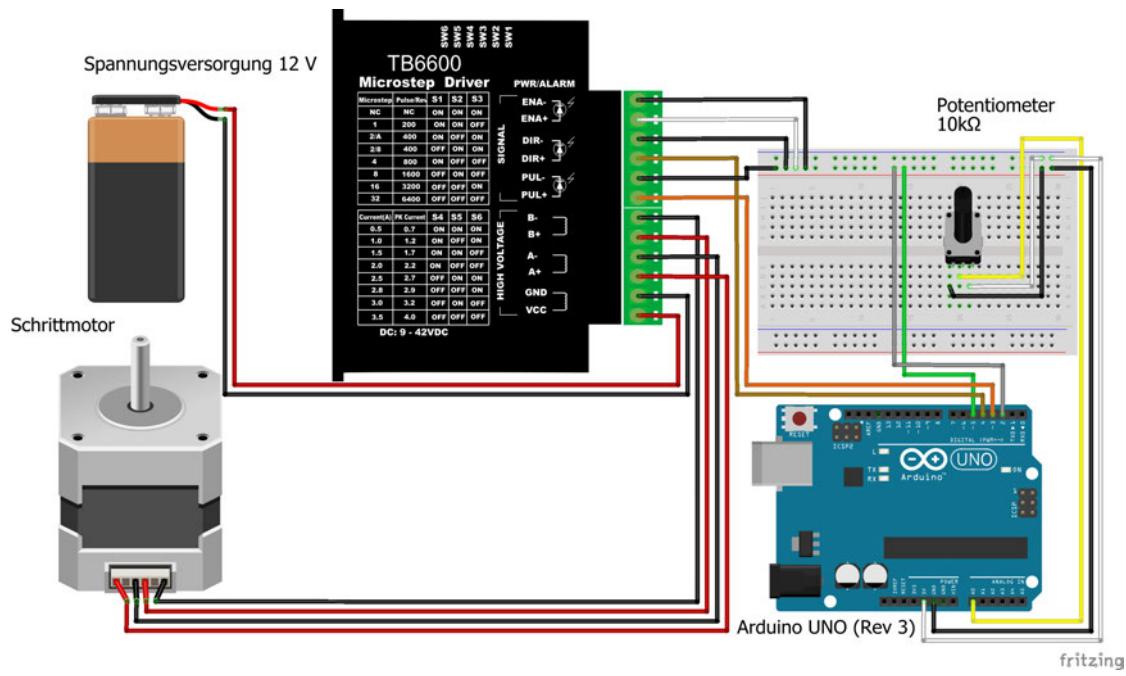


Abbildung 6.6: Aufbau zum Wickeln der Spulen

wendet. Auf der Motorwelle des Motors wird mit einer Wellenkupplung vom Typ 5x5 mm der Spulenmagnet befestigt. Der Spulendraht wird von Hand geführt. Als Strombegrenzung wird 1 A am Schrittmotortreiber eingestellt, um ein starkes Erwärmen des Schrittmotortreibers bei langer Betriebszeit zu vermeiden. Als Spannungsversorgung dient ein 12 V-Netzteil. Für die Schrittweite wird die Einstellung „Half Step“ gewählt, sodass eine volle Umdrehung 400 Schritte benötigt. Dadurch bewegt sich die Motorwelle auch bei geringer Geschwindigkeit gleichmäßig.

Über die Steckplatine lässt sich über Steckverbindungen die Drehrichtung einstellen und die Drehung unterbrechen. Die Drehgeschwindigkeit wird über ein 10 kΩ-Potentiometer eingestellt. Über die serielle Schnittstelle wird die aktuelle Rundenzahl hoch respektive herunter gezählt. Der Zähler wird per Hardware-Reset am Arduino neu gestartet. Der Aufbau zum Wickeln ist in Abbildung 6.6 zu sehen, der Programmcode des Arduino ist in der Datei „StepperTest1.ino“ auf der beigelegten CD enthalten. Es hat sich gezeigt, dass die Wellenkupplung nicht über die ganze Länge gleichmäßig anzieht, sodass der Spulenmagnet sehr locker sitzt, wenn er nur 2 mm tief in der Kupplung steckt. Das Problem kann mit einer Lage Klebeband gelöst werden, dadurch sitzt der Magnet fest genug. Da der Spulendraht per Hand geführt wird, ist es spätestens ab dem zweiten Wicklungsring

unmöglich, den Draht sauber nebeneinander zu führen. Dadurch besteht ein Wicklungsring am Ende aus deutlich weniger Wicklungslagen als in Kapitel 6.1.4 angenommen. Dementsprechend bestehen 2000 Wicklungen am Ende aus mehr als 30 Wicklungsringen und die Spule hat einen deutlich größeren Durchmesser als angenommen. Der Durchmesser für 2000 Wicklungen ist bereits so groß, dass im Tonabnehmergehäuse nicht genug Platz übrig ist, um auch noch 4000 und 6000 Wicklungen zu testen.

Beim Wickeln stellt sich heraus, dass ab ca. 1600 Wicklungen die Unwucht für einige Magneten so groß ist, dass sie sich aus der Wellenkupplung lösen und umfallen. Teilweise reißt dadurch der Spulendraht und es muss von vorne begonnen werden oder einige Wicklungen verlieren ihre Zugspannung und lösen sich vom Magneten, sodass sie sehr mühsam wieder von Hand nachgewickelt werden müssen. Insgesamt wurden zehn Spulen gewickelt um sechs brauchbare zu erhalten.

6.2.3 Wachsen

Um die Drahtlagen der Spulen zu fixieren und mögliches mikrophonisches Verhalten der Drahtlagen zu verhindern [6], wird der gesamte Tonabnehmer nach dem Wickeln und Montieren der Spulen in heißes Wachs getaucht und anschließend getrocknet. Da die Tonabnehmerspulen teilweise locker in der Bodenplatte sitzen, können sie nicht kopfüber in das heiße Wachs getaucht werden. Stattdessen werden die Spulen in das obere Teil des Gehäuses gesteckt und das Wachs mit einem entsprechend temperierten Löffel und Handschuhen in die Gehäuse geträufelt. Aufgrund der gleichen Polarisierung der Spulenmagnete stoßen sie sich teilweise so ab, dass einzelne Spulen aus ihren Steckplatz heraus gedrückt werden und am gegensätzlichen Pol des Nachbarmagneten ausrichten. Dann müssen sie vorsichtig getrennt und wieder in ihren Steckplatz gesetzt werden. Beim Wachsen baut sich eine ungleichmäßige Wachsschicht über den Spulenmagneten auf, die sich mit dem Löffel alleine nicht vollständig flüssig halten kann, um die Bodenplatte aufzudrücken. Mit einem Heißluftföhn lässt sich das erstarrte Wachs nachträglich vollflächig erwärmen und durch den Luftdruck weiter in die Gehäusekammern drücken. Dabei fließt auch viel Wachs an den äußeren Gehäuseseiten ab. Nachdem die Bodenplatte auf das flüssige Wachs gesetzt werden konnte, wird die Spaltöffnung für die Kupferlackdrähte mit einzelnen Wachsperlen gestopft und diese mit einem normalen Föhn mit weniger Luftdruck erhitzt. Ob alle Drahtlagen der Spulen vom Wachs durchdrungen wurden, lässt sich nachträglich nicht visuell überprüfen.



Abbildung 6.7: Zusammengesetzter hexaphonischer Tonabnehmer nach dem Wachsen

6.2.4 Messung des Spulenwiderstandes

Nach dem Wickeln und Wachsen wird der Spulenwiderstand gemessen, um festzustellen, ob ein Drahtbruch in einer der Spulen vorhanden ist. Dazu wird die Isolation an den Enden des Spulendrahts mit einem Messer vorsichtig freigekratzt und anschließend per Multimeter der Widerstand gemessen. Der Widerstand von fünf Spulen liegt zwischen $134,5 \Omega$ und 136Ω , die sechste Spule hat einen Widerstand von 127Ω . Das ist vermutlich eine Spule, bei der sich Wicklungen gelöst haben und manuell viele Wicklungen wieder abgetragen wurden, um die Problemstelle zu finden aber per Schätzung nachträglich nicht mehr genug Wicklungen nachgewickelt wurden. Die Messwerte liegen auf jeden Fall im erwarteten Bereich aus Gleichung 6.2.

6.2.5 Herstellen der elektrischen Verbindung

Schon bei der Messung des Spulenwiderstandes wurde festgestellt, dass sich die Lackschicht des Kupferlackdrahts schwer entfernen lässt, ohne den innen liegenden Draht mit zu beschädigen. Die offene Flamme eines Feuerzeuges reicht nicht, den Lack zu schmelzen. Damit bleibt das mechanische Abtragen als Option. Mit einer scharfen Klinge kann die Lackschicht abgetragen werden, aber das Risiko den Kupferdraht durchzutrennen ist sehr hoch. Mithilfe von Schleifpapier kann die Lackschicht gleichmäßig über die komplette Außenseite abgetragen werden, jedoch bricht der Kupferdraht leicht an den Druckstellen. Die durch die Klinge schmal abgetragenen Flächen haben zwar für die Messung des Widerstands gereicht, sind aber nicht groß genug für die Verzinnung beim Anlöten an die Stiftleisten. Stattdessen wird auf der Lochrasterplatine einer Reihe Lötösen angelötet,

die wiederum auf der Unterseite mit Silberdraht mit den Stifteleisten verbunden werden. Der Lackdraht wird mit ca. fünf Wicklungen um die Lötöse fixiert und mit viel Druck und Hitze der 330°C-heißen Lötkolbenspitze verzinnt, bis bei mindestens einer Wicklung der Lack geschmolzen ist. Die Kontaktierung wird dabei mit einer Widerstandsmessung geprüft.

6.2.6 Montage in der Gitarre

Beim Ausbau des Humbucker-Tonabnehmers hat sich gezeigt, dass dieser nicht in der Ausbuchtung der Gitarre festgeschraubt ist, sondern über eine hängende Konstruktion befestigt ist. Der Humbucker-Tonabnehmer besteht aus zwei Spulen im Single Coil-Format, die über keine Montagelaschen wie das 3D-gedruckte Gehäuse verfügen, da sie auf einem Metallblech angeschraubt sind, siehe Abbildung 6.9. Die Ausbuchtung in der Gitarre ist aber groß genug für das 3D-gedruckte Gehäuse, sodass dessen Montagelaschen nicht nachträglich abgesägt werden müssen. Die Bohrlöcher im Metallblech könnten weiter genutzt werden, indem man im 3D-gedruckten Gehäuse zwischen zwei Spulen durch die Trennwand Gewinde bohrt. Das Risiko bei den dünnen Trennwänden ist aber zu hoch, dass sie splittern und das Gehäuse bricht. Stattdessen wird der Tonabnehmer mit Klebepads auf das Metallblech geklebt. Der Streifen Lochrasterplatine muss mit Abstandshaltern auf das Blech geklebt werden, da sonst die Lötkontakte vom leitenden Blech alle kurzgeschlossen werden würden. Die abgehenden Aderleitungen werden durch vorhandene Kabelwege im Korpus der Gitarre zunächst in das Elektronikfach auf der Rückseite der Gitarre geführt.

Beim Aufkleben des Tonabnehmers auf das Blech haben sich die Bodenplatte und das Oberteil des Tonabnehmers gelöst, da das Wachs keine starke Verbindung bildet. Beim Montieren des Blechs samt Tonabnehmer und Platine hat sich gezeigt, dass das Gehäuse des Tonabnehmers doch an der oberen Kante der Ausbuchtung anschlägt und der Montagerahmen nicht über seine Bohrlöcher passt. Deswegen werden die Montagelaschen des Tonabnehmers abgesägt. Dabei ist der Tonabnehmer leider aus dem Schraubstock gerutscht und die Kupferdrähte sind beim Aufprall auf den Boden alle gerissen. Da sich schon vorher die Bodenplatte und der Oberteil des Tonabnehmers getrennt haben, wäre es eine Frage der Zeit, wann sich das Oberteil beim Kippen der Gitarre vom Blech löst und dadurch die Kupferdrähte zerreißen. Provisorisch wird der Tonabnehmer mit Kabelbindern am Blech fixiert. Von den sechs Spulen lassen sich nur zwei wieder an die

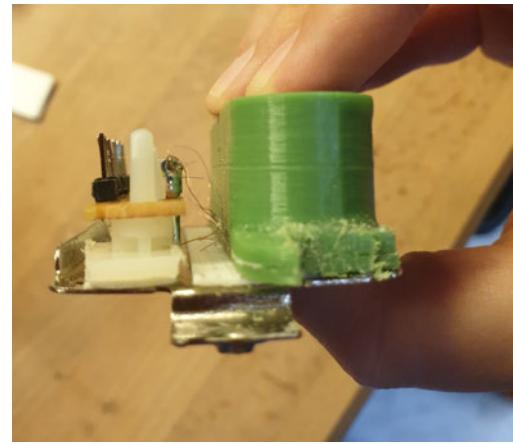
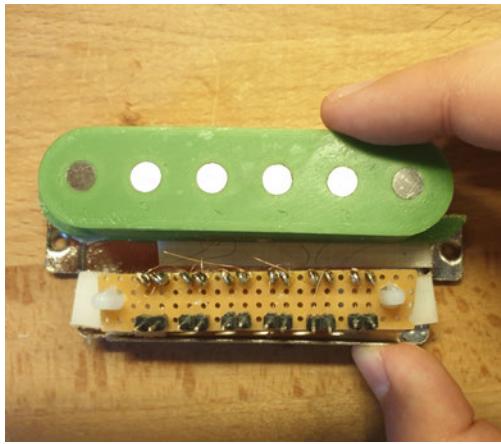
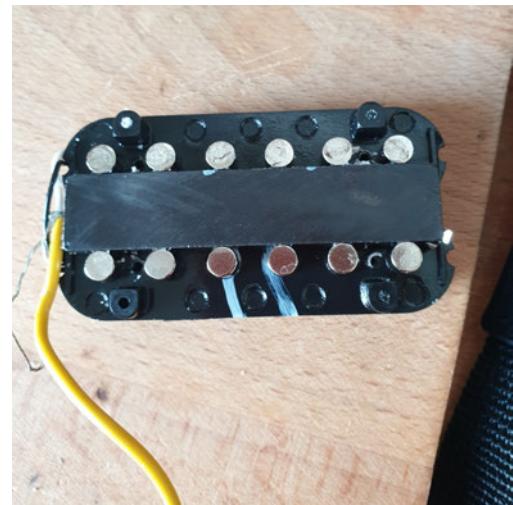


Abbildung 6.8: Hexaphonischer Tonabnehmer und Anschlussplatine auf dem Montageblech



(a) Am Montageblech fixiert



(b) Unterseite ohne Montageblech

Abbildung 6.9: Ausgebauter Humbucker-Tonabnehmer

Lötosen anlöten. Die restlichen Drähte sind zu kurz abgerissen. Damit sind aktuell nur die E- und die A-Saite aufnehmbar.

6.2.7 Kontaktweg vom Tonabnehmer zum Elektronikfach

Es werden 40 cm lange Aderleitungen vom Tonabnehmer zum Elektronikfach auf der Rückseite der Gitarre verlegt. Dafür werden die vorhandenen Kabelwege unterhalb des



Abbildung 6.10: Eingebauter hexaphonischer Tonabnehmer

hexaphonischen und des Stegonabnehmers verwendet. Für die Signale der E-Saite werden zwei rote Leitungen verwendet, für die A-Saite zwei orange, für die d-Saite zwei gelbe, für die g-Saite zwei grüne, für die h-Saite zwei blaue und für die e'-Saite zwei lila, siehe Abbildung 6.10. Beim Aufstecken der Kontakte auf die Stiftleisten hat sich gezeigt, dass die Kontaktgehäuse sehr lang sind und die Saiten leicht an den Aderleitungen anschlagen, wenn sie gespielt werden. Die Aderleitungen müssen stark gebogen werden um das zu vermeiden, wobei der ideale Biegeradius der Leitungen weit unterschritten wird. Zur Unterscheidung zwischen Masse und Signalkontakt, werden die Kontaktgehäuse der Signalleitungen mit einem Stück weißen Klebestreifen markiert. Um die Aderleitungen aus dem Elektronikfach herauszuführen wird eine Lochbohrung mit 10 mm-Durchmesser durch den Korpus neben dem vorhandenen Klinkenkontakt vorgenommen. Alternativ hätte man auch den Klinkenkontakt ausbauen können, aber dann hätte man den Stegonabnehmer nicht mehr verwenden können.

6.2.8 Messung des Amplitudengangs

Für die Messung des Amplitudengangs wird eine Messspule auf einen vom betreuenden Erstprüfer 3D-gedruckten Spulenkörper gewickelt, siehe Abbildung 6.11. Die Spule hat

einen Innendurchmesser von 6,25 mm und ist mit dem gleichen Kupferlackdraht wie die Tonabnehmerspulen gewickelt, wobei die Wicklungszahl nur 200 beträgt. Durch das Wachsen der Spule ist Wachs in den Luftkern geflossen, sodass die magnetische Feldstärke im Spulenkern etwas geringer als mit idealem Luftkern ausfällt. Der gemessene Drahtwiderstand ist 13Ω groß. Zum Anschließen wird ein Stück Lochrasterplatine unter den Spulenkörper geklebt und die Kontaktierung erfolgt wieder mit Lötösen und Stiftleisten. Zur Aufnahme des Amplitudengangs wird ein UPV Audio Analyzer im Labor für Signalverarbeitung der HAW Hamburg verwendet. Dessen Generatorausgang kann ein Sinussignal mit einer Effektivspannung im Bereich von 0,1 mV bis 10 V und steigender Frequenz erzeugen [38]. Eine frequenzabhängige Amplitudensenkung ist nicht möglich. Sein analoger Messeingang kann Eingangsspannungen im Bereich von 0,1 μ V bis 110 V erfassen. Da die Amplitude des Generatorsignals nicht frequenzabhängig gesenkt werden kann, wird anhand Gleichung 6.4 ein Vorwiderstand zur Strombegrenzung dimensioniert. Falls der Amplitudengang im Bereich von 0 Hz bis 20 kHz keine Resonanzüberhöhung mit folgendem starken Amplitudenabfall aufweist, wird die höchste Spannung bei $f = 20 \text{ kHz}$ induziert:

$$u_{ind}(t) = -N_{Spule} \cdot \int_A \mu_0 \cdot \frac{N_{Test}}{l_{Test}} \cdot \frac{U_{Gen}}{R_{vor} + R_{Test}} \cdot 2\pi f_{max} \cdot \cos(2\pi f_{max} \cdot t) \cdot d\vec{A}$$

$$\Rightarrow u_{ind,max} = -2000 \cdot \int_A 1,256 \cdot 10^{-6} \frac{N}{A^2} \cdot \frac{200}{1,5 \text{ cm}} \cdot \frac{U_{Gen}}{R_{vor} + R_{Test}} \cdot 2\pi \cdot 20 \text{ kHz} \cdot 1 \cdot d\vec{A}$$

Setzt man das Flächenintegral auf 1, setzt $U_{Gen} = 0,1 \text{ mV}$ ein und nutzt keinen Vorwiderstand, ergibt sich folgende untere Grenze für $u_{ind,max}$:

$$u_{ind,max} = -2000 \cdot 1,256 \cdot 10^{-6} \frac{N}{A^2} \cdot \frac{200}{1,5 \text{ cm}} \cdot \frac{0,1 \text{ mV}}{0 + 13 \Omega} \cdot 2\pi \cdot 20 \text{ kHz} \cdot 1$$

$$\Rightarrow u_{ind,max} = -32,4 \text{ V}$$

Sollte im Frequenzbereich von 0 Hz bis 20 kHz doch eine Resonanzüberhöhung auftreten, liegt die induzierte Spannung schnell oberhalb der zulässigen Eingangsspannung des UPV Analyzers. Deswegen wird ein Widerstand von 453Ω aus dem Materialbestand der Erstellerin dieser Arbeit in Reihe vor die Messspule gesetzt, siehe Abbildung 6.11. Das reduziert $U_{ind,max}$ auf folgenden Wert:

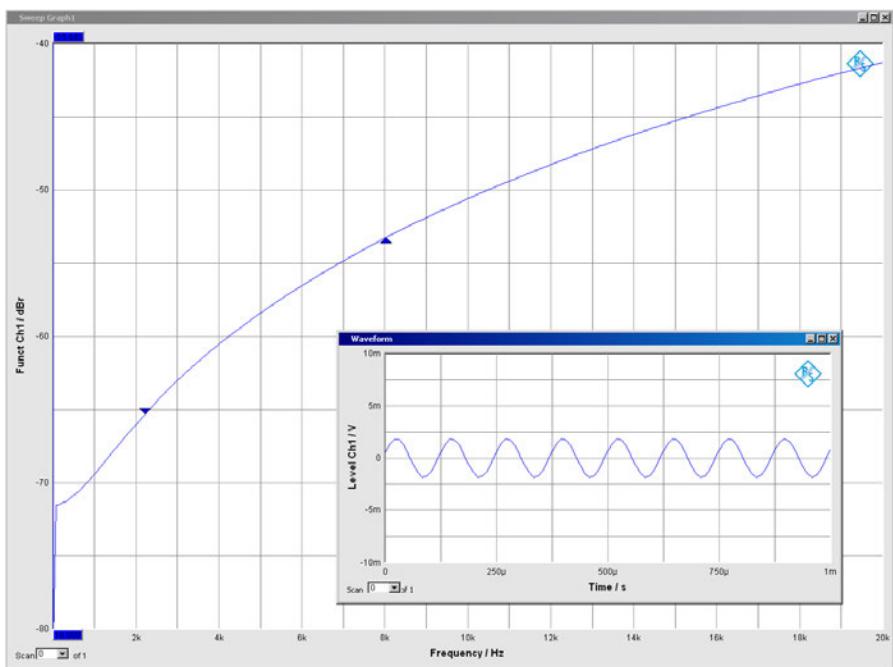
$$u_{ind,max} = -2000 \cdot 1,256 \cdot 10^{-6} \frac{N}{A^2} \cdot \frac{200}{1,5 \text{ cm}} \cdot \frac{0,1 \text{ mV}}{453 \Omega + 13 \Omega} \cdot 2\pi \cdot 20 \text{ kHz} \cdot 1$$

$$\Rightarrow u_{ind,max} = -0,9 \text{ V}$$

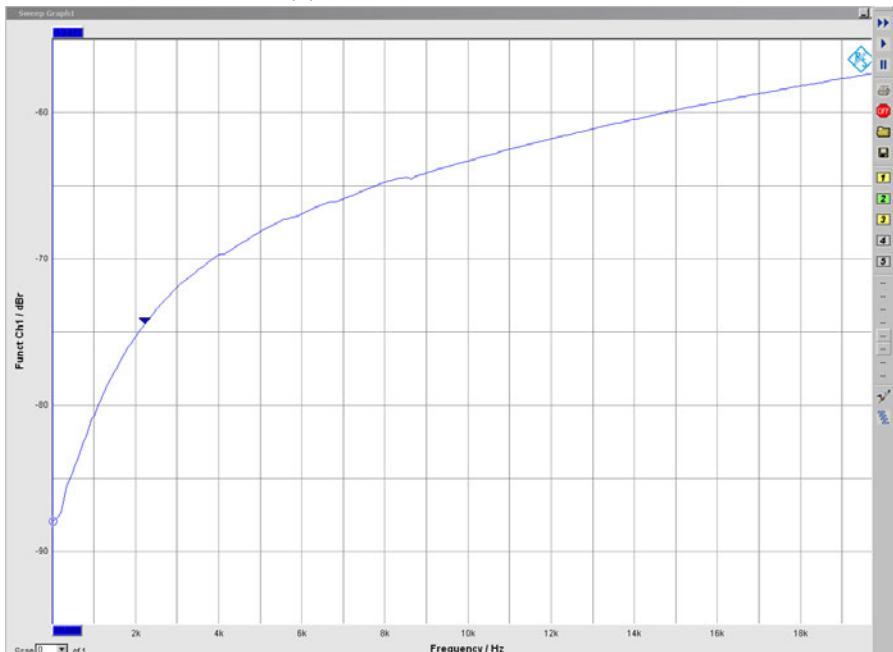


Abbildung 6.11: Messspule mit Anschlussplatine

Als Generatorsignal wird für die Messung eine Sinusspannung mit 200 mV Amplitude und konstant steigender Frequenz von 20 Hz bis 20 kHz erzeugt und an die Messspule angeschlossen. Der Messeingang des Analyzers wird an die Tonabnehmerspule für die E-Saite angeschlossen. Während der Messung wird die Messspule direkt über die Tonabnehmerspule gehalten. Der aufgenommene Amplitudengang ist in Abbildung 6.12a zu sehen. Da der Amplitudengang auf einer linearen statt auf einer logarithmierten Abszisse, wie in Abbildung 6.5 auf Seite 76, aufgetragen wird, ist kein idealer linearer Verlauf zu erwarten. Das liegt an der Darstellung der Spannung in dB: $u \text{ [dB]} = 20 \cdot \log_{10}(u \text{ [V]})$. Der Amplitudengang weist im Frequenzbereich von 0 Hz bis 20 Hz keine erkennbare Resonanzerhöhung auf, die sich dem logarithmischen Verlauf aufprägen würde. Außerdem ist kein starker Abfall des Amplitudengangs zu erkennen. Das bedeutet, dass die Resonanzüberhöhung und die darauf folgende Dämpfung sehr gering ausfallen, oder dass die Resonanzfrequenz über 20 kHz liegt. Die Startdämpfung beträgt ungefähr -72 dB, das kann daran liegen, dass das erzeugte Magnetfeld der Messspule schwächer als angenommen ist. Anschließend wird das Instrumentenkabel aus Kapitel 7 in Reihe zwischen Analyzer-Eingang und Tonabnehmerspule angeschlossen, um den Einfluss auf den Amplitudengang zu messen, siehe Abbildung 6.12b. Der relative Verlauf hat sich noch mehr einem logarithmischen Verlauf angenähert, die gesamte Messkurve wurde durch das Kabel um ca. 15,5 dB abgesenkt.



(a) Ohne Instrumentenkabel



(b) Mit Instrumentenkabel

Abbildung 6.12: Gemessene Amplitudengänge des Tonabnehmers

6.3 Weiterentwicklung des betreuenden Erstprüfers

Der betreuende Erstprüfer hat zum Spulenwickeln eine Konstruktion aus Aluminiumprofilen aufgebaut, siehe Abbildung 6.13. Sie beinhaltet eine Achse für die Kupferlackdraht-Bobine und eine Aufnahme für einen Schrittmotor, an dessen Motorwelle die Spulenmagnete mit einem zweiteiligen Gehäuse befestigt werden. Die Einzelgehäuse haben den Vorteil, dass die Spulenmagnete sicher fixiert sind und der Kupferlackdraht am oberen und unteren Ende des Gehäuses einen Anschlag beim Wickeln hat. Außerdem können die gewickelten Spulen einzeln in ein überarbeitetes Gehäuse für den hexaphonischen Tonabnehmer eingesetzt werden und können bei Bedarf ausgetauscht werden.

Zum Ansteuern des Schrittmotors wird der Schrittmotortreiber TB6600 verwendet, der durch ein Python-Programm auf einem Arduino Rev. 3 gesteuert wird. Über ein Programmfenster lassen sich die Drehgeschwindigkeit und -richtung einstellen und die aktuelle Wicklungszahl anzeigen und zurücksetzen, siehe Abbildung 6.14. Es wird mit einer Geschwindigkeit von ungefähr vier bis sechs Umdrehungen pro Sekunde gearbeitet und etwas mehr als 2000 Wicklungen auf die Spulen gewickelt. Um zu verhindern, dass die Spulen vor der Fixierung im Tonabnehmergehäuse sich gegenseitig aus ihren Steckplätzen magnetisch abstoßen, besteht jede zweite Spule aus einem Magnet mit umgekehrter Polarität und Spulendraht mit gegensinniger Wicklungsrichtung. Das sorgt dafür, dass die Ausgangsspannung aller Spulen weiterhin die gleiche Richtung aufweist. Bei der verwendeten Drahtstärke von AWG 42 beträgt der Drahtwiderstand je Spule ca. $170\ \Omega$. Nach dem Wickeln werden die Spulen bei 77°C in eine Mischung aus 80 % Paraffin und 20 % Bienenwachs getaucht, siehe Abbildung 6.15. Dafür werden sie auf einem magnetischen Teelöffel platziert und solange in das Wachs getaucht, bis keine Luftblasen mehr zwischen den Drahtwicklungen aufsteigen, um somit das Wachs alle Drahtlagen durchdrungen hat. Dabei ist zu beachten, dass das Wachs nicht zu heiß wird, um die 3D-gedruckten Einzelgehäuse der Spulen nicht zu verformen.

Für die elektrische Kontaktierung wird eine Platine entworfen, die passende Aufnahmen für die Spulenmagnete aufweist und Lötkontakte für den Spulendraht und die abgehenden Leitungen besitzt, siehe Abbildung 6.16. Die Platine beinhaltet für die Montage in der Gitarre zwei Löcher an den äußeren Enden, sodass das bisherige Montageblech des ursprünglichen Humbucker-Tonabnehmers nicht mehr benötigt wird. Die bisher verwendeten Montageschrauben werden mit Muttern an der Platine fixiert. Die einzelnen Spulen haben dasselbe Massepotential um Verkabelungsaufwand innerhalb der E-Gitarre zu reduzieren.

Da dem betreuenden Erstprüfer der Klang des ursprünglichen Halstonabnehmers der E-

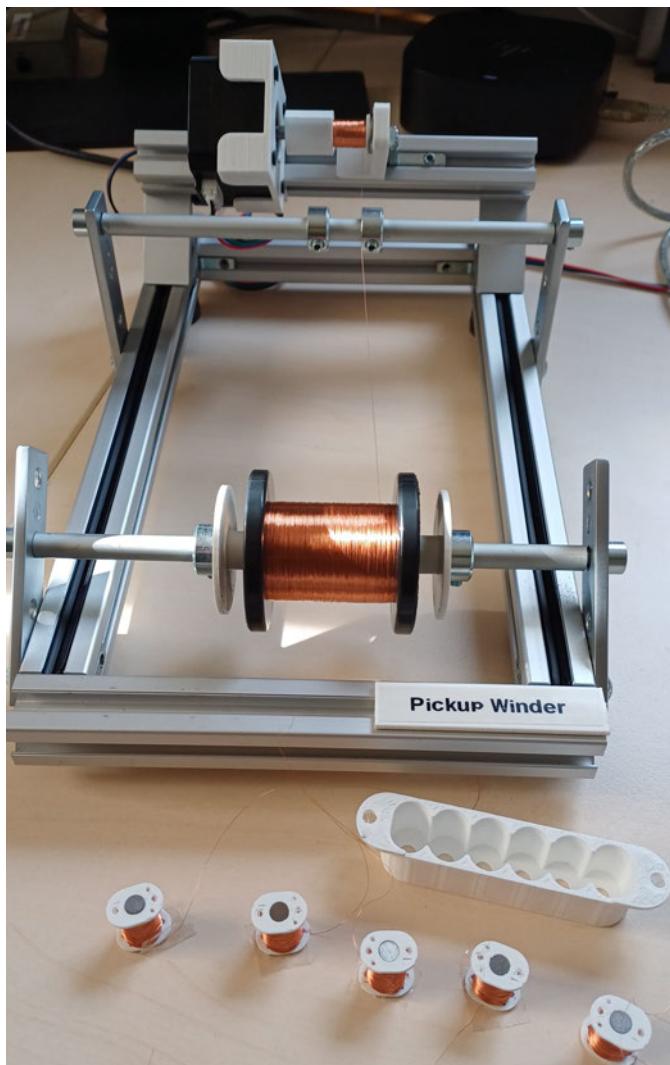


Abbildung 6.13: Wickelmaschine des betreuenden Erstprüfers

Gitarre mehr gefällt als der Klang des Stegonabnehmers, plant er den Stegonabnehmer durch seinen neuen hexaphonischen Tonabnehmer zu ersetzen und den ursprünglichen Halstonabnehmer wieder in der Gitarre zu montieren.

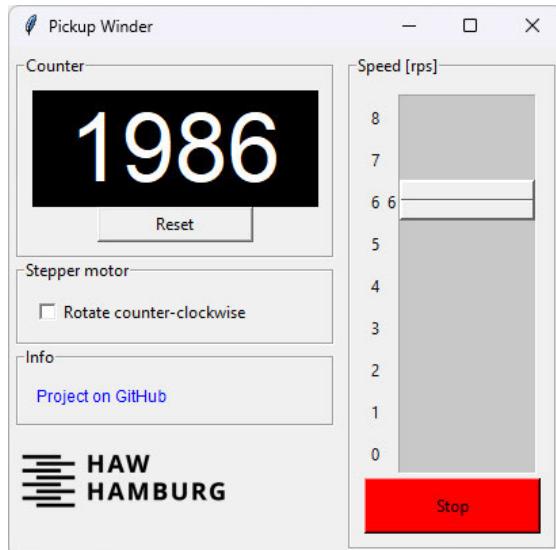


Abbildung 6.14: Python-Programmfenster zur Steuerung des Schrittmotors



Abbildung 6.15: Tonabnehmerspulen im Wachsbad

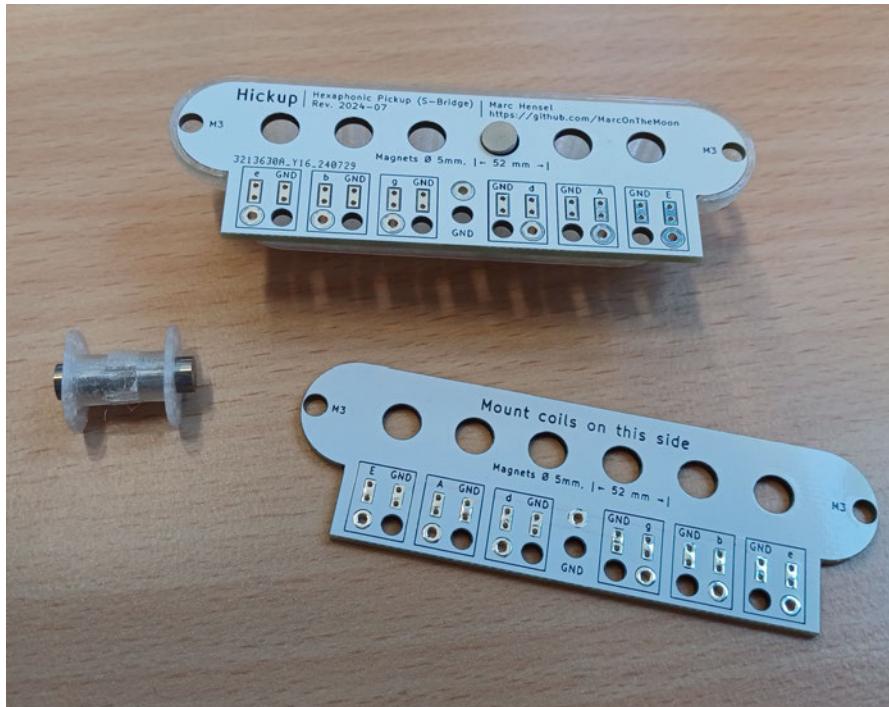


Abbildung 6.16: Bestandteile des Tonabnehmers des betreuenden Erstprüfers

7 Instrumentenkabel und Anschlusskontakte

In diesem Kapitel erfolgt die Konzipierung und Umsetzung des Instrumentenkabels und der benötigten Anschlusskontakte, um es an der E-Gitarre anschließen zu können.

7.1 Konzept

7.1.1 Aufbau des Kabels und des Übertragungsweges

Beim hexaphonischen Tonabnehmer liegen sechs eigenständige Tonsignale vor, die sich bei der Übertragung nicht gegenseitig durch Übersprechen beeinflussen sollen. Deswegen soll jedes Tonsignal gegen die anderen Signale abgeschirmt werden. Außerdem benötigt das Audio-Interface aus Kapitel 5.2.3 sechs einzelne Kabel mit 1/4 in-Klinkensteckern auf der Eingangsseite. Statt ein großes Kabel mit sechs Koaxialleitern aufwändig aufzuspalten, werden von Anfang an sechs einzelne Instrumentenkabel eingesetzt. So sind die Tonsignale untereinander durch eine solide Abschirmung geschützt.

7.1.2 Auswahl des Kabelmaterials

Instrumentenkabel kann man einerseits fertig konfektioniert für bestimmte Längen kaufen oder individuell selber anfertigen. Dazu gibt es Instrumentenkabel als Meterware, sodass man nur noch die Anschlusskontakte anlöten muss. Für die sechs Instrumentenkabel für den hexaphonischen Tonabnehmer ist es wichtig, dass die Kabelkapazität so gering wie möglich ausfällt und die Abschirmung geflochten aufgebaut ist, wie in Kapitel 2.8 erläutert. Außerdem dürfen die Kabel nicht zu dick sein, damit sie weiterhin mit 1/4 in-Klinkensteckern konfektioniert werden können.

7.1.3 Aufbau der Übergabeeinheit

Um die Aderleitungen auf der Seite des Tonabnehmers mit den Klinkenkontakten des Instrumentenkabels verbinden zu können, wird eine Übergabeeinheit benötigt, siehe Abbildung 7.1. Diese muss an der Gitarre montierbar sein. Dafür wird ein kleines Gehäuse erstellt, dass auf der einen Seite zwei DIN-Einbaubuchsen und auf der anderen Seite sechs Klinkeneinbaubuchsen hat. Die Aderleitungen des Tonabnehmers werden auf der freien Kontaktseite auf zwei DIN-Stecker geführt, sodass die Verbindung zur Übergabeeinheit trennbar wird. Innerhalb der Übergabeeinheit werden die DIN-Einbaubuchsen und die Klinkeneinbaubuchsen mit Aderleitungen direkt verbunden, siehe Abbildung 7.2. Die gesamte Übergabeeinheit ist zu groß und sperrig, um sie an die Gitarre selbst zu montieren, deswegen wird eine Montage am Gitarrentgurt vorgesehen. Dafür ist für die Übergabeeinheit einen Führungsring auf der Unterseite des Gehäuses vorgesehen, durch den der Gitarrentgurt gefädelt wird.

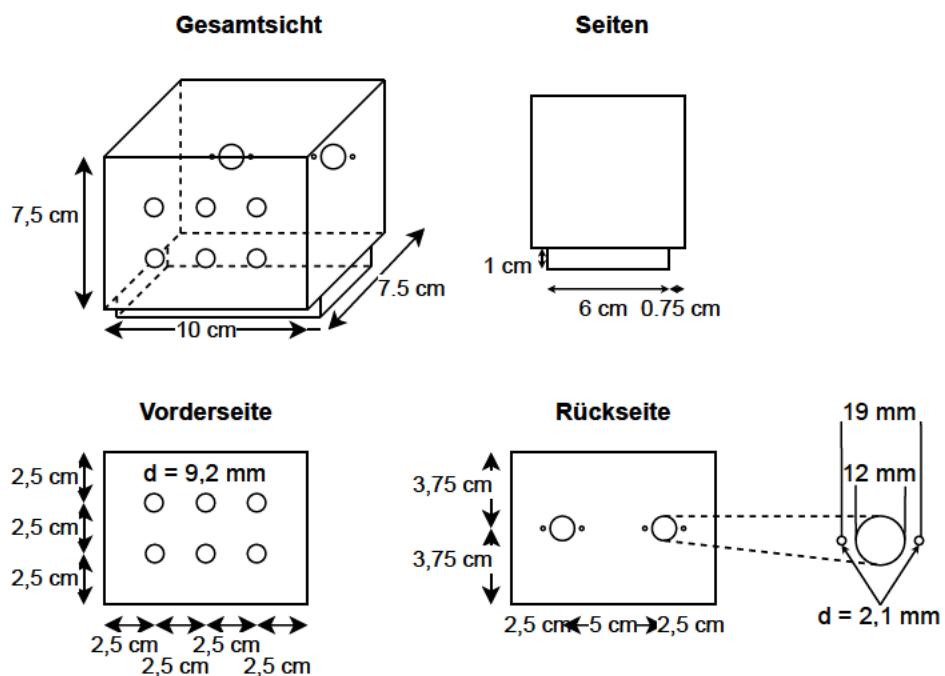


Abbildung 7.1: Entwurf der Übergabeeinheit

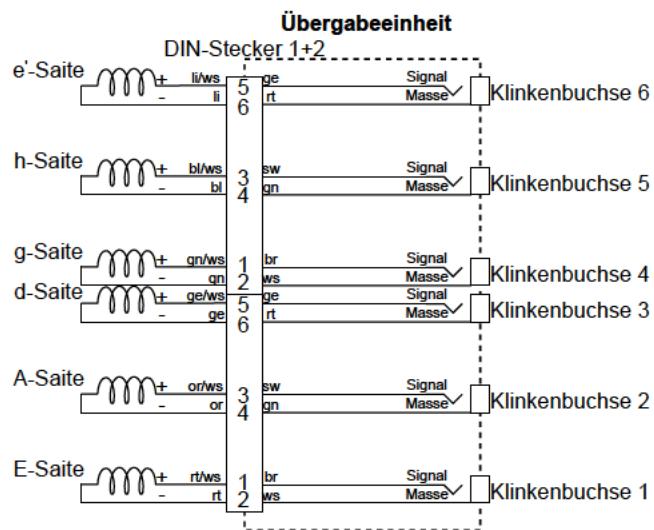


Abbildung 7.2: Kontaktwege innerhalb der Übergabeeinheit

7.2 Umsetzung

7.2.1 Instrumentenkabel

Für das Instrumentenkabel wird das Modell „the sssnake MPP8030 Snake“¹ beschafft, das über acht Instrumentenkabel in einem gemeinsamen Mantel und fertig konfektionierte 1/4 in-Kontakte verfügt. Zur Kabelkapazität gibt es keine Angaben. Die einzelnen Kabel sind mit Farbringern in rot, orange, gelb, grün, blau, lila, weiß und grau an den jeweiligen Kontakten und mit Nummern auf dem jeweiligen Kabelmantel beschriftet.

7.2.2 Aufbau der Übergabeeinheit

Da zum aktuellen Zeitpunkt der Arbeit nur noch zwei Spulen des Tonabnehmers funktionieren, wird zunächst eine kleinere Übergabeeinheit vom betreuenden Erstprüfer mit einem 3D-Drucker gedruckt, die jeweils drei Tonabnehmersignale übertragen kann. Sie hat eingangsseitig eine DIN-Einbaubuchse mit sechs Kontakten und ausgangsseitig drei Klinkeneinbaubuchsen, siehe Abbildung 7.3. Um die Kontaktwege einheitlich zu halten,

¹https://www.thomann.de/de/the_sssnake_sk4213_snake_klinke.htm

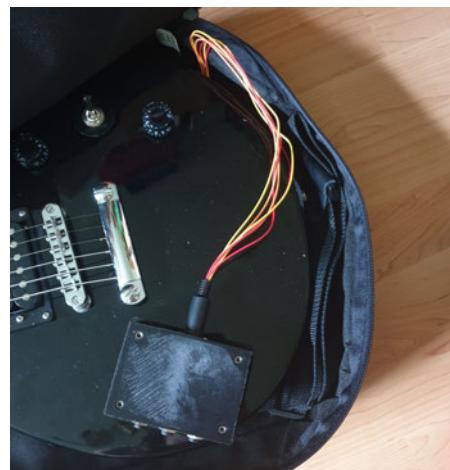


Abbildung 7.3: Übergabeeinheit an der Gitarre angeschlossen

werden wieder jeweils zwei Aderleitungen in den Farben Rot bis Lila verwendet. Da aktuell nur ein DIN-Stecker mit sechs Kontakten verwendet wird, werden zunächst die roten, orangen und gelben Aderleitungen angelötet. Die Kontaktbelegung für den DIN-Stecker ist in Tabelle 7.1 zu sehen. Die Klinkeneinbaubuchsen werden mit roten, orangen und gelben Klebepunkten markiert.

Tabelle 7.1: Kontaktbelegung des DIN-Steckers (von vorne betrachtet)

	Kontakt 1	Kontakt 2	Kontakt 3	Kontakt 4	Kontakt 5	Kontakt 6
Position	unten links	unten rechts	mitte links	mitte rechts	oben links	oben rechts
Farbe	rot/weiß	rot	orange/weiß	orange	gelb/weiß	gelb
Funktion	Signal E	Masse E	Signal A	Masse A	Signal d	Masse d

8 Tonhöhenanalyse und MIDI-Konvertierung

In diesem Kapitel erfolgt die Tonhöhenanalyse mittels YIN-Algorithmus und die anschließende Konvertierung in MIDI-Dateien.

8.1 Aufzeichnung der Gitarrentöne

Für die Aufzeichnung der Gitarrentöne werden die zwei funktionierenden Tonabnehmerspulen mit der Übergabeeinheit und dem Instrumentenkabel an das Audointerface angeschlossen. Das Audointerface wird über eine USB 2.0 Schnittstelle an einem Computer mit Windows-Betriebssystem Version 10 angeschlossen. Am Audointerface werden auf der Vorderseite die Eingänge 1 und 2 mit Instrumenten-Pegel verwendet und außerdem die Eingangsverstärkung für beide Eingänge an ihren Reglern bis zum Anschlag rechts aufgedreht. Über die Leuchten „SIG“ und „CLIP“ wird angezeigt, ob ein Eingangssignal erkannt wird und ob der Pegel größer als der maximal erlaubte Eingangspegel ist, siehe Abbildung 8.1. Nach der Installation des Gerätetreibers für das Audointerface können seine Eingänge im Audio-Editor als Signalquelle ausgewählt werden. Da nur zwei Tonabnehmerspulen funktionieren, werden im Audio-Editor lediglich Stereo-Dateien mit einer



Abbildung 8.1: Einstellungen am Audointerface für die Eingänge 1 und 2



Abbildung 8.2: Aufgenommene Signale der E- und A-Saite für einen Ton

Abtastrate von 44,1 kHz und einer Buffergröße von 2048 Samples aufgenommen. Kanal 1 entspricht dabei den Signalen der E-Saite und Kanal 2 entspricht den Signalen der A-Saite. Die Aufnahme der Gitarren-Samples erfolgt durch den Bruder der Autorin dieser Arbeit, der über 13 Jahre Spielerfahrung verfügt. Die Autorin selbst spielt zum aktuellen Stand der Arbeit nur Bass-Gitarre, die eine andere Spieltechnik als E-Gitarren voraussetzt. Für die Gitarren-Samples wird die Gitarre im Sitzen gespielt, da die Montagevorrichtung der Übergabeeinheit noch nicht vorliegt und so verhindert wird, dass die Übergabeeinheit beim Spielen im Stehen an ihren Kabeln herabhängt. Die Töne werden mit einem Plektrum angeschlagen und es wird auf sonstige Spieltechniken, siehe Kapitel 2.4, verzichtet. Die Gitarre wird vor dem Spiel mit einem Stimmgerät des Typs „PW-CT-12“ gestimmt. Das Stimmgerät wird am Kopf der Gitarre angeklemmt und misst über die Vibration des Korpus die Tonhöhe und zeigt diese auf einem Display an. Als Referenz wird a' mit 440 Hz eingestellt. Nach der Aufnahme der Samples wird vorhandene Stille vor dem Anschlag des ersten Tons entfernt und die Daten als WAV-Datei abgespeichert, siehe Abbildung 8.2. Es hat sich gezeigt, dass bei maximaler Verstärkung durch das Audiointerface die Tonabnehmerspule der A-Saite höhere Spannungen erzeugt als die Tonabnehmerspule der E-Saite. Das führt dazu, dass Töne, die auf der E-Saite gespielt werden, von der A-Tonabnehmerspule lauter aufgenommen werden als von der E-Tonabnehmerspule. Um das zu kompensieren, wird die Verstärkung am Audiointerface für die A-Tonabnehmerspule so weit zurück gedreht, bis Töne der E-Saite

von der E-Tonabnehmerspule lauter aufgenommen werden, aber Töne der A-Saite weiterhin von der A-Tonabnehmerspule lauter aufgenommen werden. Die aufgezeichneten WAV-Dateien sind alle im Order „Gitarrensamples“ auf der beigelegten CD zu finden.

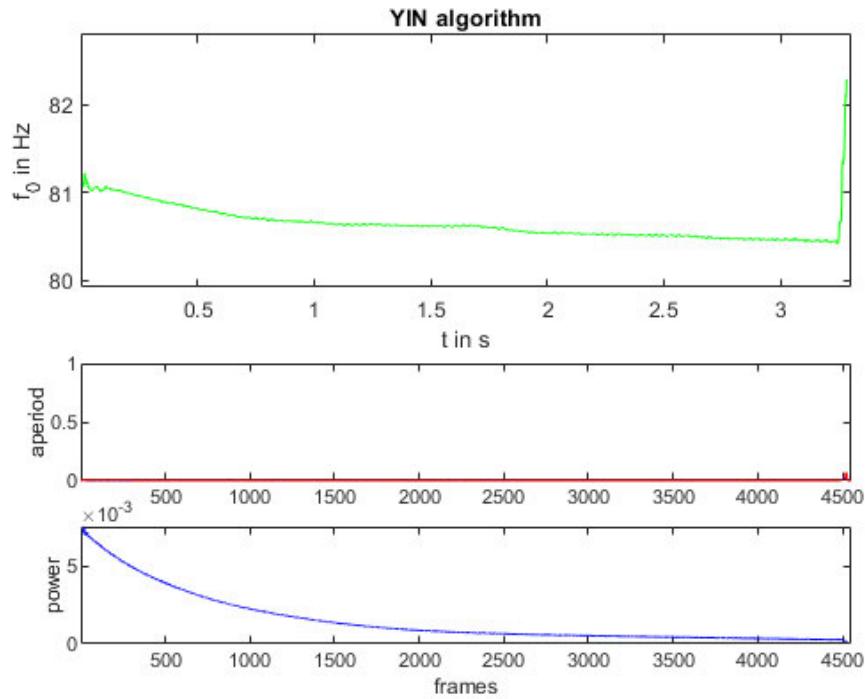
8.2 Tonhöhenanalyse

8.2.1 Aufbau des Programms

Wie in Kapitel 5.2.6 beschrieben, wird die Implementierung des YIN-Algorithmus verwendet. Die Dateien dafür liegen im Ordner „yin“ auf der beigelegten CD. Aus Performancegründen liegen die fundamentalen Funktionen des Algorithmus als externe mex-Dateien, geschrieben in C, vor, die abhängig von der genutzten Programmierumgebung vorkompiliert werden müssen. Befolgt man die Installationshinweise hierzu, kann das YIN-Programm unter GNU Octave 9.1.0 lauffähig gemacht werden.

Die Datei *testyin.m* dient als Startdatei. Übergibt man dort der Funktion *yin* eine WAV-Datei, erzeugt die Funktion einen Plot der ermittelten Frequenzen innerhalb der Datei und zeigt zusätzlich die ermittelte Aperiodizität und Signalleistung an, siehe Abbildung 8.3. Die Aperiodizität wird in zwei Farben, rot und blau, dargestellt. Die blaue Kurve stellt eine feiner aufgelöste Version der gemessenen Aperiodizität in rot dar. Die blaue Kurve ist nur dann zutreffend, wenn ein valides Minimum der Differenzfunktion ermittelt wurde. In der Konsole wird für die gesamte Datei die geringste Aperiodizität und der dazugehörige Zeitpunkt angegeben und wie der ermittelte Ton heißt. Diese Information ist nur für Dateien, die einen einzigen gespielten Ton enthalten, aussagekräftig. Beim Tonnamen wird zusätzlich die Größe Cent angegeben, dies dient einer genaueren Auflösung. Dabei besteht der Abstand zwischen zwei Halbtönen aus 100 Cent [12]. Im Beispiel aus Abbildung 8.3 liegt der gemessene Ton zwischen den Tönen E und Es. In der Funktion *yin* werden die zu analysierenden Daten und ihre Abtastrate ausgelesen, falls eine WAV-Datei übergeben wurde, und die optional übergebenen Analyseparameter überprüft. Anschließend wird der gesamte Datensatz samt Analyseparametern an die Funktion *yink* übergeben, siehe Abbildung A.1 auf Seite 161 im Anhang. Nachdem *yink* ausgeführt wurde, werden in *yin* die Ergebnisplots dargestellt, falls kein Rückgabeparameter angegeben wurde. Sonst werden keine Plots erzeugt und die Ergebnisse samt verwendeter Analyseparameter als Struktur zurückgegeben.

In *yink* werden die Daten bufferweise an die Subfunktion *yin_helper* übergeben, siehe Abbildung A.2 auf Seite 162 im Anhang. Die Daten werden zunächst mit einem Tiefpass



```
>> testyin
best: 80.4797 Hz (E2 -41 cents) at 3.0643s (aperiodic/total power: 0.00023128)
```

Abbildung 8.3: Beispieldaten der Funktion *yin*

gefiltert. Anschließend werden die externen mex-Funktionen zur Durchführung des YIN-Algorithmus aufgerufen, siehe Abbildung A.3 auf Seite 163 im Anhang. Innerhalb des Datenbuffers werden die Daten fensterweise analysiert und f_0 , die Aperiodizität und die Signalleistung ermittelt. Diese Fenstergröße muss dabei mindestens zwei Signalperioden der längsten erwarteten T_0 groß sein, siehe Kapitel 2.11. Ist ein Fenster fertig analysiert, wird es um eine bestimmte Schrittweite weitergeschoben, sodass sich die Fenster auch überlappen können. Das erhöht die zeitliche Auflösung der Ergebnisse.

8.2.2 Übergabeparameter

Die Funktion *yin* benötigt die zu analysierenden Daten und akzeptiert optional eine Struktur mit Analyseparametern oder eine Angabe zur Abtastrate als zweiten Übergabeparameter.

Eingangsdaten

Die zu analysierenden Daten können die Dateiendung .wav, .macsnd, .aiff/.aifc, .nist, .wff und .csv haben. Außerdem werden reine ASCII- und Binärdateien akzeptiert. Zuletzt können auch Datenmatrizen übergeben werden. Bei ihnen muss die zugrundeliegende Abtastfrequenz übergeben werden, entweder alleine als zweiter Übergabeparameter oder als Feld in der Struktur der Analyseparameter.

Analyseparameter

Die Analyseparameter müssen als Struktur aufgebaut sein und sind optional. Werden keine oder nur einzelne Parameter übergeben, erstellt *yin* eine eigene Struktur mit Standardwerten oder ergänzt eine übergebene Struktur. Die Felder der Struktur werden im Folgenden erläutert.

- SR: Abtastrate, mit der die Daten aufgenommen wurden. Muss angegeben werden, wenn sie nicht aus den Daten oder der übergebenen Datei ermittelt werden kann.
- min. f_0 : geringste erwartete Frequenz, standardmäßig 30 Hz.
- max. f_0 : höchste erwartete Frequenz, standardmäßig SR/4.
- thresh: Schwellwert für die minimal erlaubte Aperiodizität bei der Suche nach T_0 , standardmäßig 0,1.
- relflag: gibt an, ob thresh relativ zum globalen Minimum der Differenzfunktion oder zu 0 gilt, standardmäßig 1 (thresh ist relativ zum Minimum).
- hop: Schrittweite, mit dem das Analysefenster über die Daten geschoben wird, standardmäßig 32 Samples.
- range: Samplebereich, der analysiert werden soll, standardmäßig alle Samples.

- wsize: Größe des Analysefensters, standardmäßig SR/min. f_0 .
- bufsize: Größe des Samplebuffers, der an Subroutinen übergeben wird, nicht zu verwechseln mit wsize. Standardmäßig 10000.
- lpf: gibt die Grenzfrequenz an, mit der die Daten eingangs durch einen Tiefpass gefiltert werden. Standardmäßig SR/4.
- shift: gibt an, wie bei der Differenzfunktion die Daten zueinander verschoben werden. 0 bedeutet, dass die Datensätze symmetrisch zueinander verschoben werden, 1 bedeutet, dass ein Datensatz unverändert bleibt und der zweite nach rechts verschoben wird, -1 bedeutet, dass ein Datensatz unverändert bleibt und der zweite nach links verschoben wird. Standardmäßig 0.
- plotthreshold: Schwellwert, ab wann Werte für f_0 nicht mehr genutzt werden, da die dazugehörige Aperiodizität zu hoch ist. Standardmäßig 0,2.

8.2.3 Ergänzung der Amplitudenbestimmung

Für die MIDI-Konvertierung wird neben der Tonhöhe und Tonlänge auch die Lautstärke benötigt. Daher wird in der Funktion *yin_helper* die Amplitudenbestimmung ergänzt. Dafür wird für jedes Analysefenster die maximale Signalamplitude bestimmt und zurückgegeben. Außerdem wird in der Funktion *yin* der Plot um den Verlauf der Amplitude ergänzt, siehe Abbildung 8.4. Es ist zu erkennen, dass die Signalamplitude und die Signalleistung einen ähnlichen Verlauf aufweisen, wobei die Signalleistung eine glattere Kurve beschreibt. Der Wertebereich der Signalamplitude liegt höher als der für die Signalleistung.

8.2.4 Analyse statischer Töne

Zunächst werden einzelne Töne auf der E- und A-Saite angeschlagen, separat als WAV-Datei abgespeichert und mit dem YIN-Programm analysiert. Dabei wird zunächst die Tonspur der Tonabnehmerspule der jeweiligen Saite verwendet und die andere ignoriert. Dazu werden in der Datei *testyin* die WAV-Dateien mit der Funktion *audioread* als zweispaltige Matrix eingelesen und jeweils die erste oder zweite Spalte an *yin* übergeben. Für die folgende Analyse werden diese Analyseparameter verwendet:

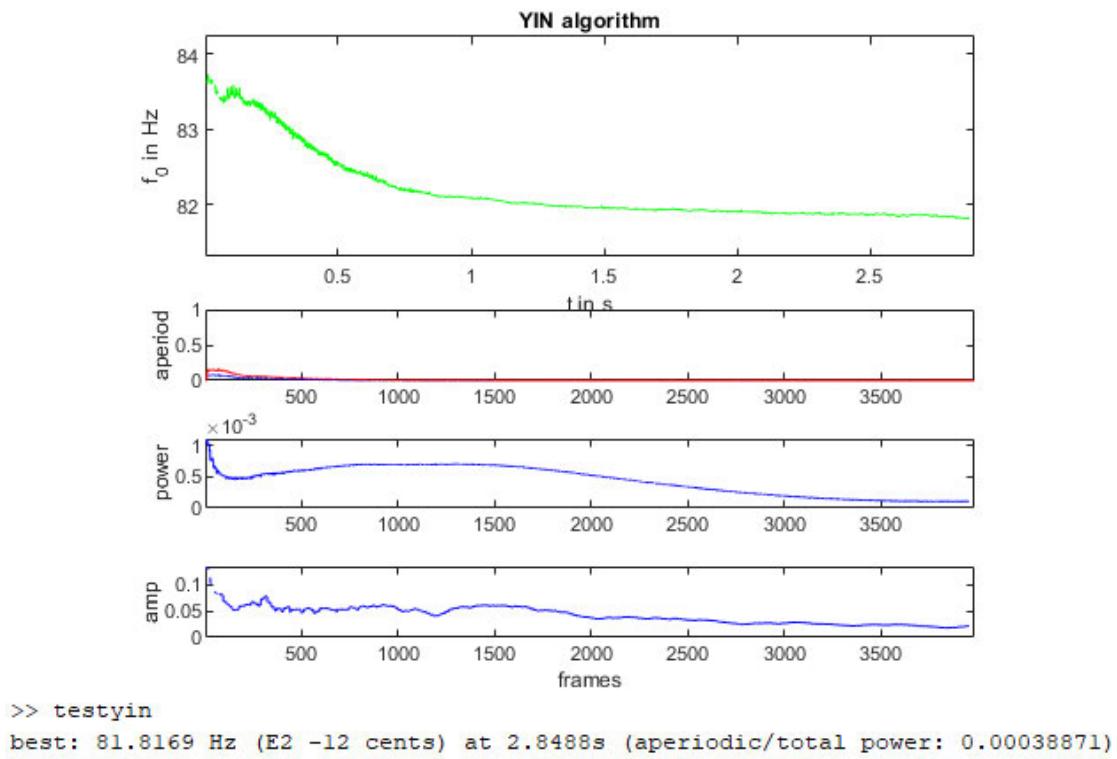


Abbildung 8.4: Ausgabe der Funktion *yin* ergänzt um die Signalamplitude

- min. f_0 : 80 Hz, da E eine Frequenz von 82,406 Hz hat.
- max. f_0 : 1175 Hz, da der höchste spielbare Ton bei 22 Bünden auf der e'-Saite d''' mit 1147,7 Hz ist.
- SR: 44,1 kHz, übernommen aus den Angaben aus Wavosaur.

Die restlichen Parameter verbleiben bei ihren Standardwerten.

Leere E-Saite: Das Ergebnis von *yin* ist in Abbildung 8.4 zu sehen. Es ist zu erkennen, dass f_0 über dem erwarteten Wert von 82,4 Hz beginnt und dann kontinuierlich sinkend sich 82,4 Hz annähert, je länger der Ton ausklingt. Für die geringste Aperiodizität zum Zeitpunkt 2,849 s wird die Tonhöhe E mit -12 Cents angegeben, das heißt die Tonhöhe ist niedriger als der erwartete Wert. Bei tiefen Tönen werden kleine Abweichungen der Frequenz stärker wahrgenommen als bei hohen Tönen, da der Abstand zwischen tiefen

Tönen geringer ist als bei hohen Tönen, weil die Abstände zwischen Tönen logarithmisch zunehmen. Der starke Abfall von f_0 und der Signalleistung lässt sich mit dem Anschlag mit dem Plektrum erklären. Mit einem Plektrum wird die Saite kurzzeitig stark ausgelenkt, bevor die gleichmäßige Schwingung einsetzt. Zupft man die Saite mit dem Finger an, wird sie nicht so stark ausgelenkt und der Verlauf von f_0 und der Signalleistung würde flacher beginnen.

F auf der E-Saite: In Abbildung A.7 auf Seite 167 ist zu erkennen, dass der Verlauf für f_0 deutlich über dem erwarteten Wert von 87,307 Hz beginnt, aber beim Ausklingen des Tons sich dem Wert nähert. Das bestätigt auch die Tonhöhenangabe für die geringste Aperiodizität zum Zeitpunkt 1,9839 s.

A auf der E-Saite: In Abbildung A.8 auf Seite 168 ist zu erkennen, dass der Verlauf für f_0 wieder über dem erwarteten Wert von 110 Hz beginnt, aber beim Ausklingen des Tons sich dem Wert nähert. Das bestätigt auch die Tonhöhenangabe für die geringste Aperiodizität zum Zeitpunkt 1,346 s.

Ais auf der E-Saite: In Abbildung 8.5 ist ein Oktavfehler zu erkennen. Die erwartete Frequenz für Ais liegt bei 116,56 Hz und wird auch im Großteil der Datei erkannt. Zwischen 0,9 s und 1,39 s werden Frequenzen um 232 Hz erkannt, was gerade dem 1. Oberton von Ais entspricht.

d auf der E-Saite: In Abbildung 8.6 ist zu erkennen, dass der gespielte Ton stark um die erwartete Frequenz 146,83 Hz herum schwankt und die geringste Aperiodizität zum Zeitpunkt 0,03 s sehr früh vorliegt. Für geübte Hörende ist das Schwanken der Tonhöhe auch in der WAV-Datei zu hören.

e auf der E-Saite: In Abbildung A.9 auf Seite 168 ist zu erkennen, dass der Ton e richtig erkannt wird und es sonst keine relevanten Auffälligkeiten gibt.

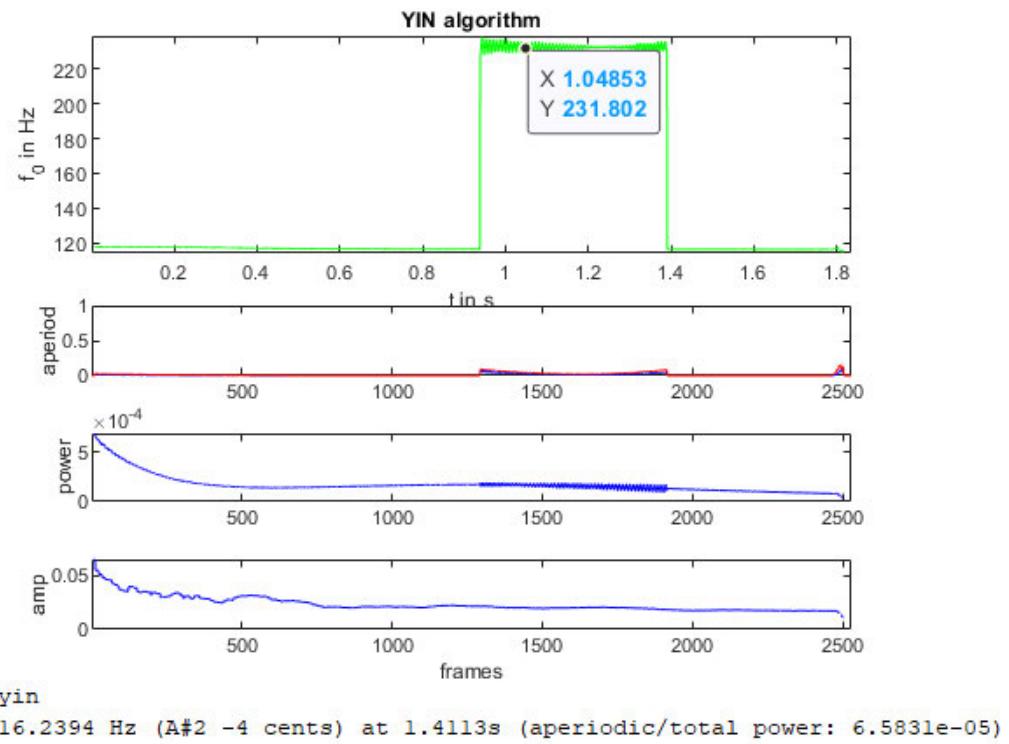


Abbildung 8.5: Ausgabe der Funktion *yin* für den Ton Ais auf der E-Saite

d' auf der E-Saite: In Abbildung 8.7 ist zu erkennen, dass der Ton gegen Ende der Aufnahme geringfügig um die erwartete Frequenz von 293,66 Hz herum schwankt. Das kann daran liegen, dass es der höchste spielbare Ton auf der E-Saite ist und schwierig sauber klingend zu spielen ist.

Als Nächstes werden Töne auf der A-Saite analysiert und die zweite Datenspalte an *yin* übergeben.

Leere A-Saite: In Abbildung A.10 auf Seite 169 ist zu erkennen, dass der Ton A richtig erkannt wird und es keine relevanten Auffälligkeiten gibt.

Ais auf der A-Saite: In Abbildung A.11 auf Seite 169 ist zu erkennen, dass beim Tonanschlag f_0 kurz von 117,5 Hz auf 120 Hz springt. f_0 erreicht gegen Ende aber die erwartete Frequenz von 116,56 Hz.

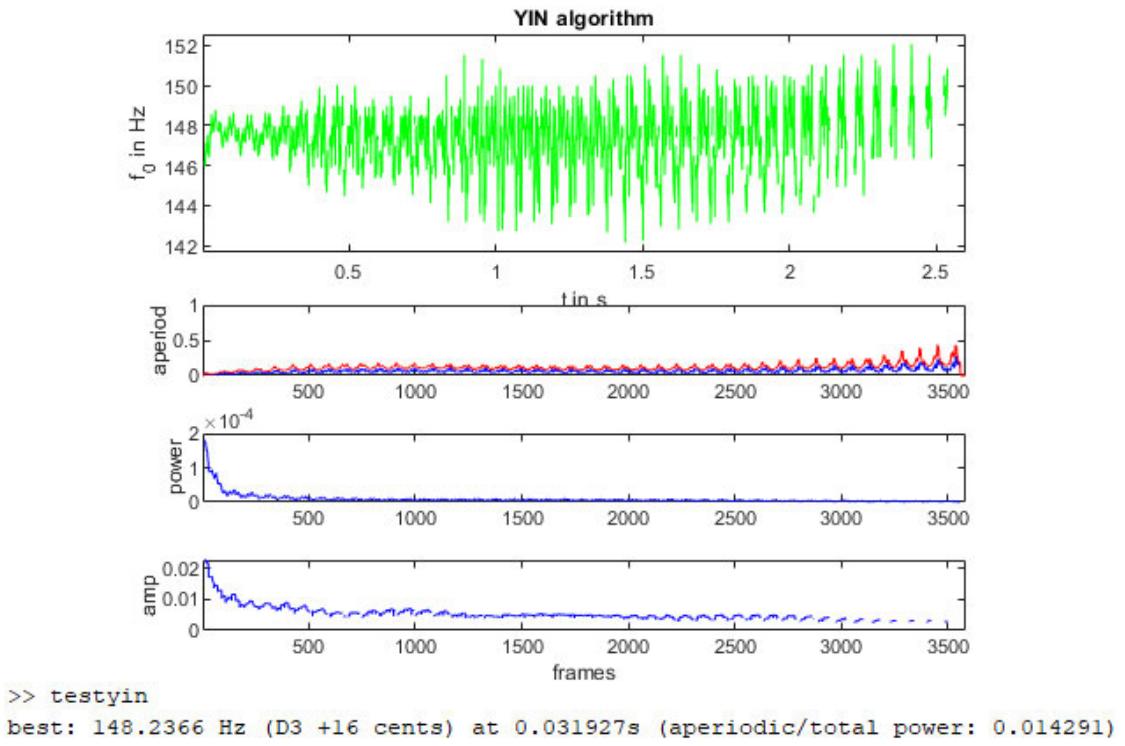


Abbildung 8.6: Ausgabe der Funktion *yin* für den Ton d auf der E-Saite

cis auf der A-Saite: In Abbildung 8.8 ist erneut ein Oktavfehler zu erkennen. Im Bereich zwischen 0,8 s und 1,2 s werden Frequenzen um 277 Hz erkannt, was dem 1. Oberton der erwarteten Frequenz von 138,59 Hz entspricht.

a auf der A-Saite: In Abbildung A.12 auf Seite 170 ist zu erkennen, dass der Ton a richtig erkannt wird und es keine relevanten Auffälligkeiten gibt.

g' auf der A-Saite: In Abbildung A.13 auf Seite 170 ist zu erkennen, dass der Ton g' als höchster spielbarer Ton auf der A-Saite korrekt erkannt wird. Im Gegensatz zu d' auf der E-Saite sind keine Schwankungen um die erwartete Frequenz zu erkennen.

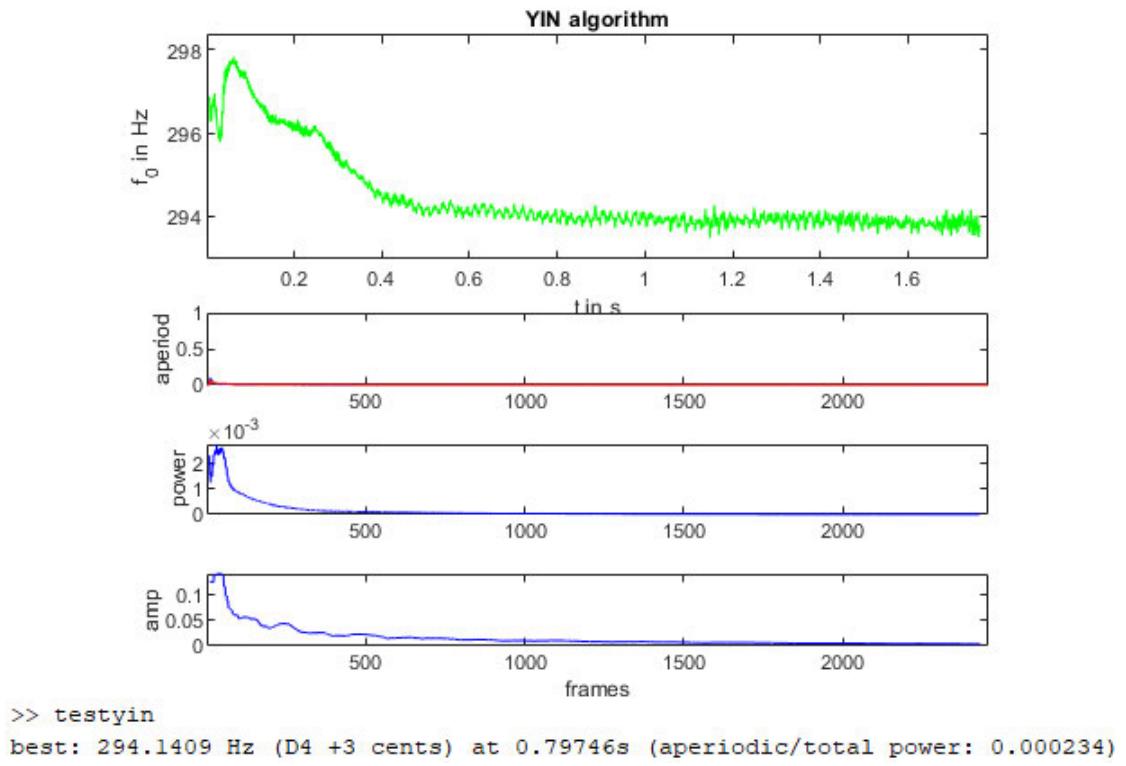


Abbildung 8.7: Ausgabe der Funktion *yin* für den Ton *d'* auf der E-Saite

8.2.5 Analyse mehrerer Töne auf einer Saite

Bei der Analyse von Tonfolgen ist es wichtig, dass sowohl die Tonhöhen als auch die Tonlängen und Pausenlängen korrekt erkannt werden. Dafür werden zunächst auf der E- und A-Saite nacheinander verschiedene Töne als Viertel bei ca. 60 BPM gespielt, das heißt jeder Ton ist ungefähr 1 s lang. Als „plotthreshold“ wird der Wert 0,15 verwendet.

Viertel auf der E-Saite: Folgende Töne werden gespielt: E, F, c, H, cis, A, E, F, G, A, E. In Abbildung 8.9 ist zu erkennen, dass die Aperiodizität bei Tonwechseln stark ansteigt und auch die Signalleistung und -amplitude durch den neuen Tonanschlag einen Sprung aufweisen. Der Wert der Aperiodizität hilft maßgeblich bei der Feststellung von Tonwechseln. Die Werte für f_0 entsprechen den erwarteten Frequenzbereichen. Lediglich für den ersten gespielten Ton A ist bei $t = 6$ s ein Oktavfehler zu erkennen. Der Graph von f_0 beschreibt diskrete Tonstufen außer beim Übergang von F auf c bei $t = 2$ s und von cis

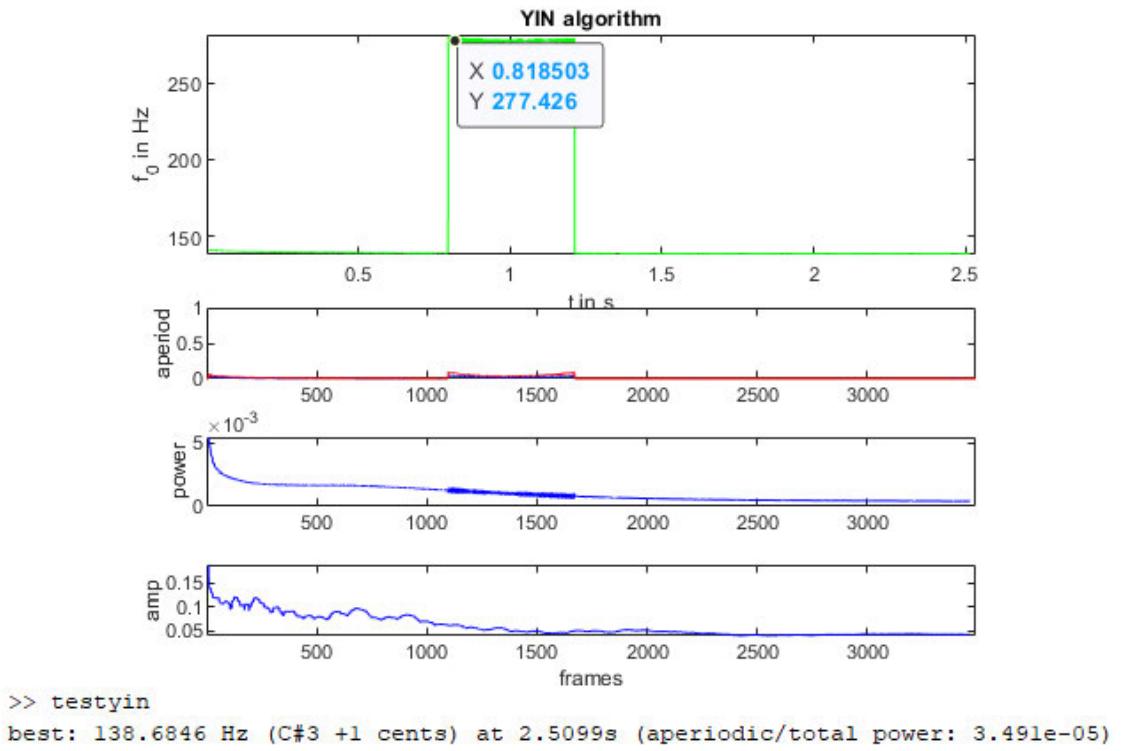


Abbildung 8.8: Ausgabe der Funktion *yin* für den Ton cis auf der A-Saite

auf A bei $t=4,7$ s. Das liegt daran, dass die Spielhand auf dem Griffbrett umgesetzt wurde und dadurch eine Art Slide, siehe Kapitel 2.4, zu hören ist. Die kurzen Unterbrechungen im Graph von f_0 röhren vom Schwellwert „plotthreshold“ für die Aperiodizität. Liegt die Aperiodizität über diesem Wert, werden die Schätzungen für f_0 ignoriert, da sie zu aperiodisch sind. Setzt man diesen Schwellwert niedriger, werden die Abstände zwischen einzelnen Tönen größer, aber Frequenzsprünge beim Tonwechsel, wie beim Flankenübergang von $t=7,2$ s, werden reduziert. Dies hat aber den Nachteil, dass besonders kurze Töne ignoriert werden können. Setzt man den Schwellwert sehr hoch, werden die Töne länger und es treten mehr Frequenzsprünge im Bereich der Tonwechsel auf. Dies ist in Abbildung 8.10 deutlich zu erkennen, wenn der Schwellwert auf 0,7 erhöht wird.

Viertel auf der A-Saite: Folgende Töne werden gespielt: A, H, c, d, e, f, g, a. In Abbildung 8.11 auf Seite 109 ist wieder ein stufenartige Verlauf von f_0 zu erkennen. Diesmal treten keine Obtavfehler auf. Zum Zeitpunkt $t=4,7$ s werden Frequenzen im

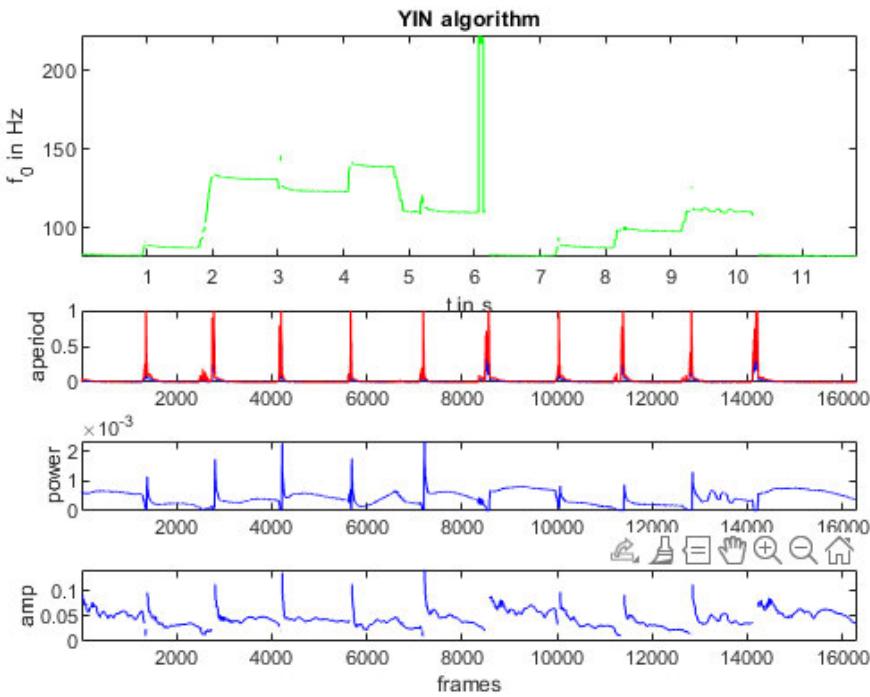


Abbildung 8.9: Ausgabe der Funktion *yin* für Vierteltöne auf der E-Saite

Bereich 330 Hz geschätzt, obwohl diese Frequenz kein Oberton der Töne f oder g ist. Betrachtet man die Signalleistung zu dem Zeitpunkt, ist zu erkennen, dass gar kein Ton klingt, da die Spielhand in dem Augenblick versetzt wurde. Das heißt die Frequenzschätzung hat Frequenzen in der Stille erkannt, die gar nicht gespielt wurden. Setzt man einen zusätzlichen Schwellwert für die Signalleistung, kann man solche Pausen besser erkennen. Dieser Schwellwert wird zunächst auf 3,6 % der maximalen Signalleistung gesetzt. Liegt die Leistung darunter, liegt kein gespielter Ton vor und die Frequenzschätzungen für diesen Bereich werden ignoriert. Dies ist in Abbildung 8.12 auf Seite 110 dargestellt. Es ist zu erkennen, dass die Frequenzschätzungen zum Zeitpunkt $t=4,7$ s nicht mehr aufgeführt werden. Andererseits wird der Ton H zum Zeitpunkt $t=0,9$ s nun unterbrochen, da er sehr leise klingt.

Achtel auf der A-Saite: Es wird 16 mal die leere A-Saite als Achtel bei ca. 120 BPM gespielt, das heißt ein Ton ist ungefähr 250 ms lang. In Abbildung 8.13 auf Seite 111 ist zu erkennen, dass 16 Töne gespielt werden, aber die Töne so kurz sind, dass nur die An-

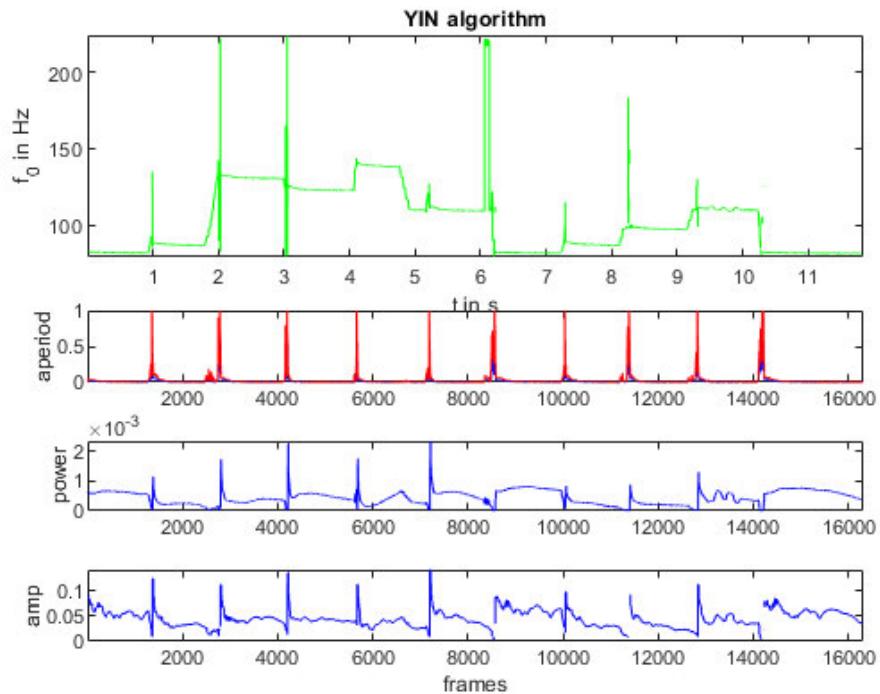


Abbildung 8.10: Ausgabe der Funktion *yin* für Vierteltöne auf der E-Saite mit „plotthreshold“=0,7

schlagimpulse zu erkennen sind. Sie klingen nicht lang genug, um einen geraden Verlauf der Zielfrequenz annehmen zu können. Die Töne befinden sich dennoch nahe der erwarteten Frequenz von 110 Hz. Außerdem ist zu erkennen, dass jeder zweite Impuls mit einer tieferen Frequenz beginnt als der vorherige. Das liegt daran, dass die Töne mit einem Wechselschlag, siehe Kapitel 2.4, gespielt wurden. Da die Töne kontinuierlich ohne explizite Pause gespielt werden, treten bei den Tonwechseln starke Frequenzspitzen auf, da die aktuelle Saitenschwingung durch einen neuen Anschlagimpuls überlagert wird. Zum Vergleich wird die Sequenz ein weiteres Mal ohne Plektrum mit Fingerstyle aufgenommen, siehe Abbildung 8.14 auf Seite 112. Diesmal sind die Tonverläufe deutlich gerader und ohne erkennbaren Anschlagimpuls. Dafür treten bei einigen Tonwechseln weiterhin Frequenzspitzen auf. Außerdem ist zu erkennen, dass die Signalleistung und -amplitude um mehr als die Hälfte geringer ausfallen, da die Töne beim Fingerstyle tendenziell leiser klingen als beim Spiel mit dem Plektrum.

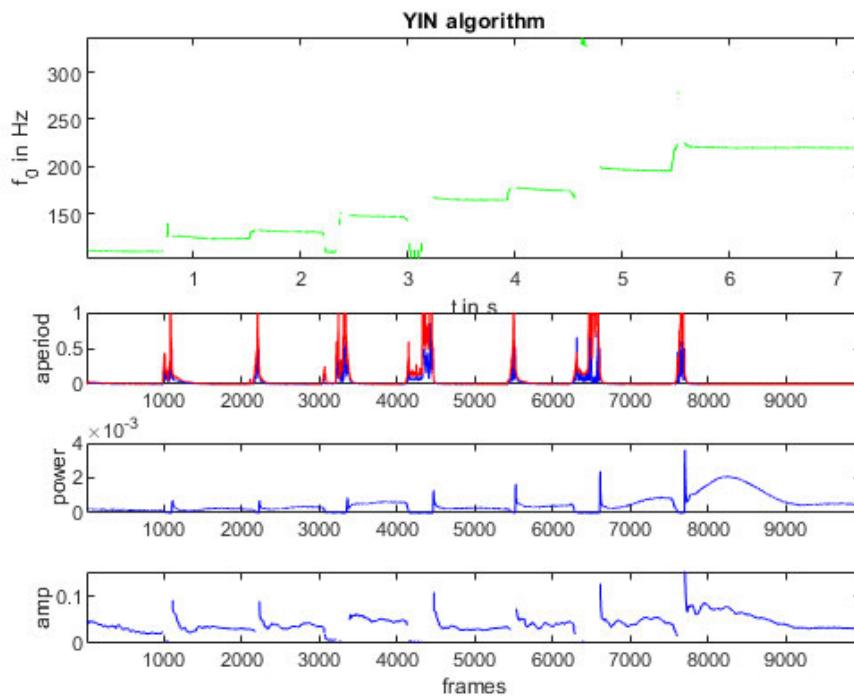


Abbildung 8.11: Ausgabe der Funktion *yin* für Vierteltöne auf der A-Saite

8.2.6 Interferenz zwischen zwei Tonabnehmerspulen

Nun wird die jeweils andere Tonspur der bisherigen Aufnahmen betrachtet, um zu sehen, wie viel Signalleistung die Nachbarspule aufgenommen hat.

Leere E-Saite: Für den Ton E auf der E-Saite wird das aufgenommene Signal der A-Tonabnehmerspule analysiert. Es ist zu erkennen, dass der Verlauf von f_0 in Abbildung A.14 auf Seite 171 dem Verlauf von f_0 in Abbildung 8.4 auf Seite 101 ähnelt. Lediglich die Signalleistung und -amplitude fallen geringer aus.

Ais auf der E-Saite: Für den Ton Ais auf der E-Saite wird das aufgenommene Signal der A-Tonabnehmerspule analysiert. Es ist zu erkennen, dass der Verlauf von f_0 in Abbildung A.15 auf Seite 171 dem Verlauf von f_0 in Abbildung 8.5 auf Seite 103 ähnelt, wobei in Abbildung A.15 kein Oktavfehler auftritt. Außerdem ist am Ende von des Verlaufes

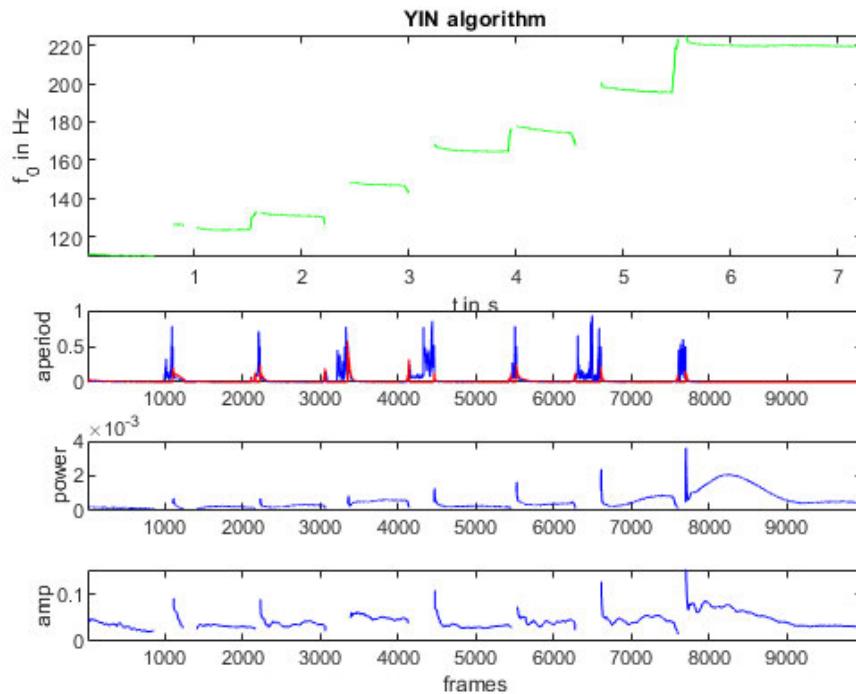


Abbildung 8.12: Ausgabe der Funktion *yin* für Vierteltöne auf der A-Saite mit Leistungsschwellwert=3,6 % der maximalen Leistung

von f_0 ein steiler Anstieg der Frequenz zu erkennen, obwohl kein Ton mehr klingt. Das liegt daran, dass die Signalleistung zu dem Zeitpunkt bei 10 % der Maximalleistung und damit immer noch über den Schwellwert von 3,6 % aus Kapitel 8.2.5 liegt, sodass die Werte für f_0 noch ausgegeben werden.

Leere A-Saite: Für den Ton A auf der A-Saite wird das aufgenommene Signal der E-Tonabnehmerspule analysiert. Es ist zu erkennen, dass der Verlauf von f_0 in Abbildung A.16 auf Seite 172 dem Verlauf von f_0 in Abbildung A.10 auf Seite 169 ähnelt. Lediglich die Signalamplitude hat in Abbildung A.16 einen höheren Startwert.

cis auf der A-Saite: Für den Ton cis auf der A-Saite wird das aufgenommene Signal der E-Tonabnehmerspule analysiert. Es ist zu erkennen, dass der Verlauf von f_0 in Abbildung A.17 auf Seite 172 dem Verlauf von f_0 in Abbildung 8.8 auf Seite 106 bis zum Zeitpunkt $t=1,25$ s ähnelt. Danach hört der Graph für f_0 in Abbildung A.17 auf, da

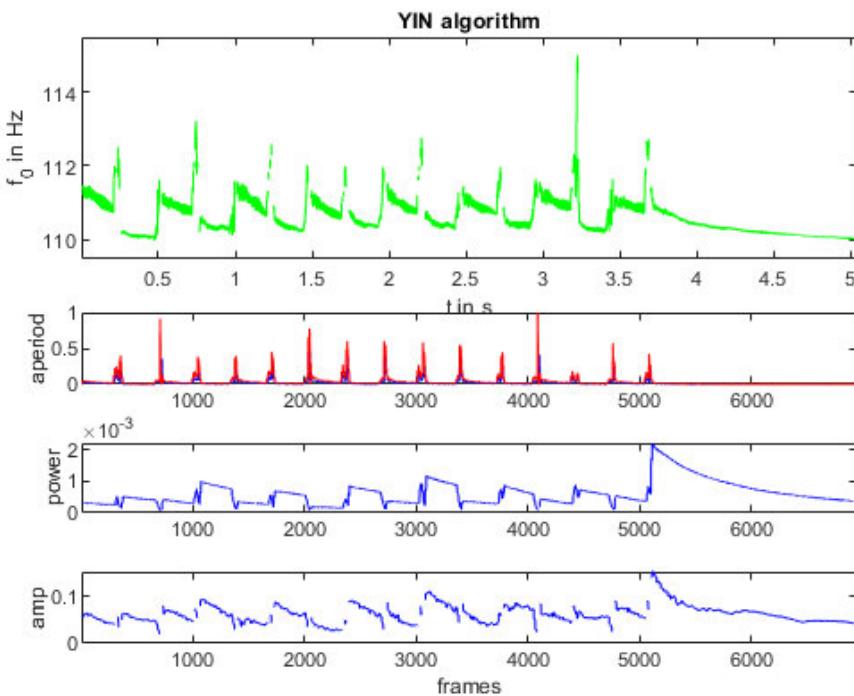


Abbildung 8.13: Ausgabe der Funktion *yin* für Achteltöne auf der A-Saite

die Signalleistung den Schwellwert von 3,6 % unterschritten hat. Außerdem haben die Signalleistung und -amplitude generell wieder einen niedrigeren Wertebereich.

8.2.7 Analyse mehrere Töne auf allen Saiten

Es wird die A-Moll-Pentatonik bestehend aus den Tönen A, c, d, e, g, a, c', d', e', g' und a' als Viertel über alle Saiten der E-Gitarre bei ca. 60 BPM gespielt. Die Töne A und c werden auf der E-Saite gespielt und die Töne d und e auf der A-Saite. In Abbildung 8.15 ist zu erkennen, dass die E-Tonabnehmerspule die Töne A und c und den Anschlagimpuls der Töne d und e aufgenommen hat. Für die höheren Töne ist die Signalleistung zu gering. In Abbildung 8.16 ist zu erkennen, dass die A-Tonabnehmerspule die Töne A, c, d und e aufgenommen hat. Für die Töne A und c, aufgenommen mit der E-Tonabnehmerspule, sind die Signalleistungen aber höher als bei der Aufnahme mit der A-Tonabnehmerspule.

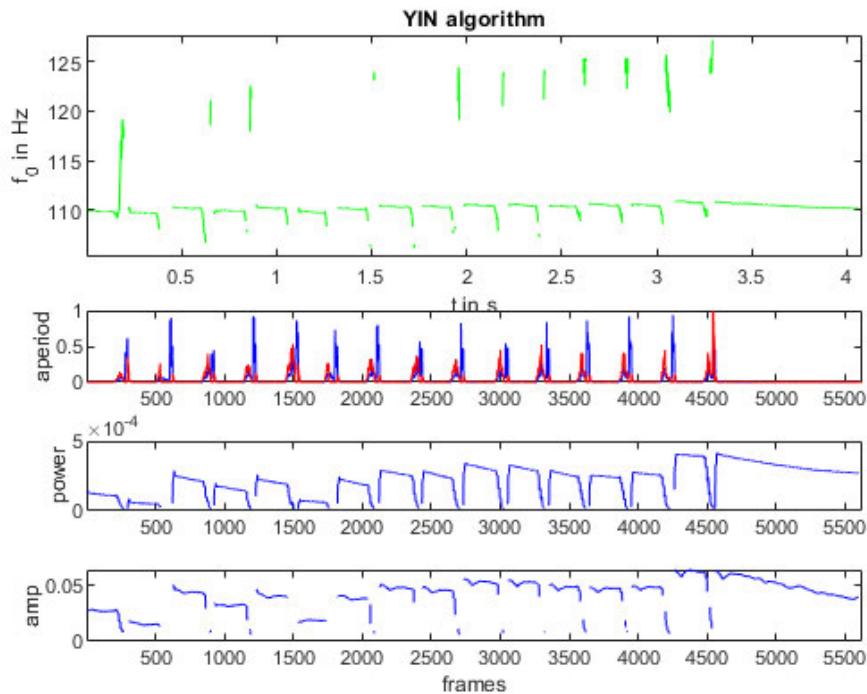


Abbildung 8.14: Ausgabe der Funktion *yin* für Achteltöne auf der A-Saite gespielt mit Fingerstyle

8.2.8 Analyse von Intervallen auf der E- und A-Saite

Als Nächstes werden zwei Töne auf der E- und A-Saite gleichzeitig angeschlagen und überprüft, ob das Programm den auf der jeweiligen Saite klingenden Ton zuordnen kann.

A auf der E-Saite und dis auf der A-Saite: In Abbildung 8.17 auf Seite 115 ist für die E-Tonabnehmerspule zu erkennen, dass der Graph von f_0 deutliche Schwankungen um die Zielfrequenz von 110 Hz aufweist. Das Gleiche gilt für die Signalleistung und -amplitude. In Abbildung 8.18 auf Seite 116 ist für die A-Tonabnehmerspule zu erkennen, dass die rote Kurve für die Aperiodizität fast vollständig über den bisherigen Schwellwert von 0,15 verläuft, weswegen dieser auf 0,5 gesetzt wird. Der Verlauf von f_0 weist ähnliche Schwankungen wie in Abbildung 8.17 auf. Obwohl auf der A-Saite der Ton dis mit einer Frequenz von 155,5 Hz gespielt wurde, werden nur Frequenzen zwischen 110,2 Hz und

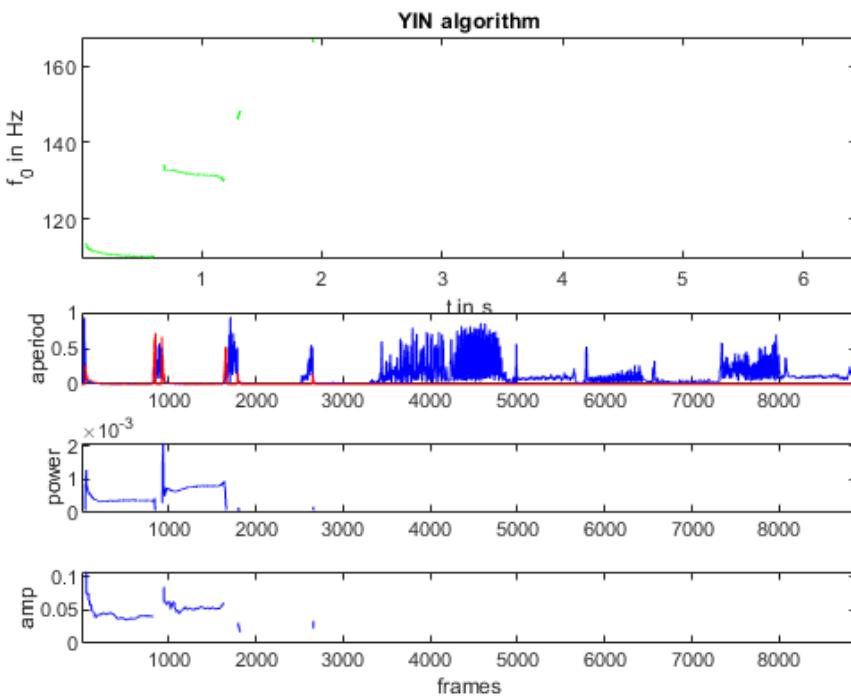


Abbildung 8.15: Ausgabe der Funktion *yin* für die A-Moll-Pentatonik aufgenommen mit der E-Tonabnehmerspule

102,5 Hz erkannt. Das entspricht einem Ton zwischen A und Gis. Die ermittelte Signalleistung für die A-Saite sinkt deutlich schneller ab als die Signalleistung für die E-Saite.

a auf der E-Saite und g auf der A-Saite: In Abbildung 8.19 auf Seite 117 ist für die E-Tonabnehmerspule zu erkennen, dass der Graph von f_0 Schwankungen um die Zielfrequenz von 220 Hz aufweist, wobei die Hüllkurve zwischen 223 Hz und 217 Hz liegt. Die Aperiodizität weist beim Tonanschlag eine Spitze um 0,5 auf und ist danach sehr gering. Für die Aufnahme der A-Saite wird der „plotthreshold“ wieder von 0,15 auf 0,5 erhöht, da die Aperiodizität für den Ton g beim Tonanschlag über dem vorherigen Schwellwert von 0,15 liegt. In Abbildung 8.20 auf Seite 118 ist für die A-Tonabnehmerspule zu erkennen, dass der Graph für f_0 um die Zielfrequenz 195,9 Hz in einem Bereich von 191 Hz bis 203 Hz schwankt. Die Signalleistung und -amplitude weisen auch einen schwankenden Verlauf auf.

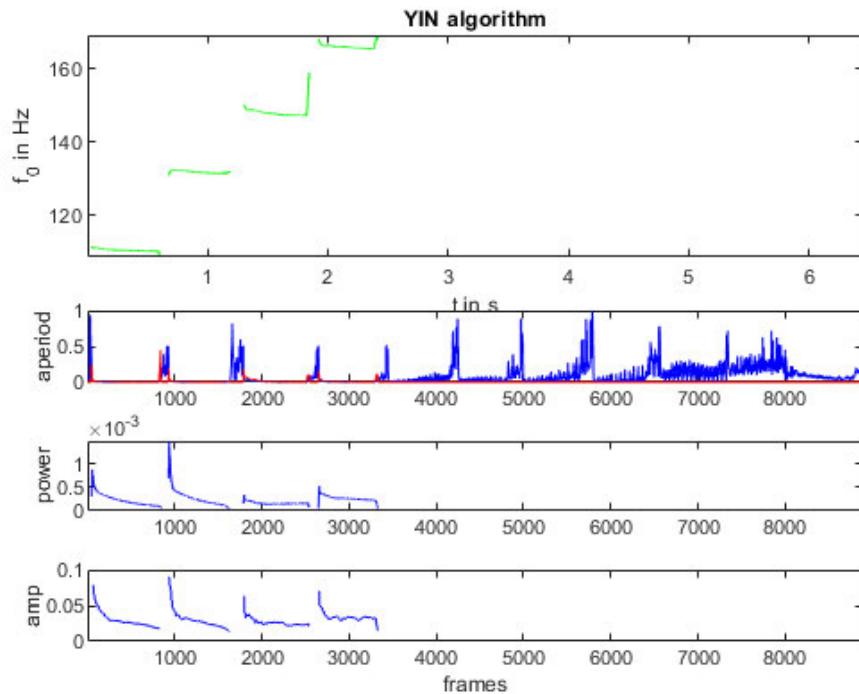


Abbildung 8.16: Ausgabe der Funktion *yin* für die A-Moll-Pentatonik aufgenommen mit der A-Tonabnehmerspule

Intervall mit Tonwechsel: Es werden die Töne H auf der E-Saite und fis auf der A-Saite gleichzeitig angeschlagen. Anschließend wird der Ton e auf der A-Saite angeschlagen während der Ton H auf der E-Saite weiter klingt. In Abbildung 8.21 auf Seite 119 ist für die E-Tonabnehmerspule zu erkennen, dass beim ersten Intervall der Graph für f_0 eine geringere Schwankung um die erwartete Frequenz aufweist als beim Anschlag von e ab $t=1,1$ s. Ab dann nimmt f_0 Werte zwischen 129 und 119 Hz an, was schon im Wertebereich der Nachbartöne Ais und c liegt. Unabhängig davon ist die Aperiodizität sehr gering, abgesehen von den Tonwechseln. Für die Abbildung 8.22 auf Seite 120 wurde für die Aufnahme der A-Tonabnehmerspule der „plotthreshold“ wieder auf 0,5 erhöht, da die Aperiodizität für den Ton fis deutlich über dem vorherigen Schwellwert von 0,15 liegt. Die Töne fis und e werden erkannt und die Schwankungen um die erwartete Frequenz fallen bei beiden Tönen geringer aus als für den Ton H. Kurz zu Beginn der Aufnahme wird auch der Ton H erkannt. Das liegt daran, dass mit einem Plektrum unmöglich zwei Saiten zeitgleich angeschlagen werden können, sondern diese nacheinander angeschlagen

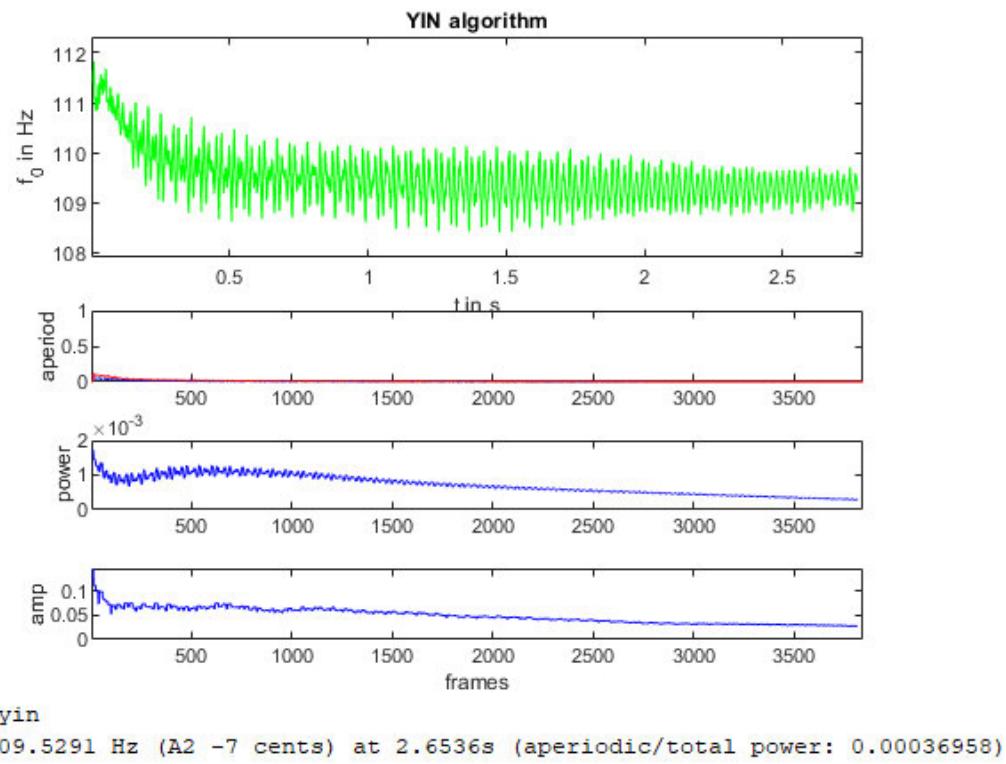


Abbildung 8.17: Ausgabe der Funktion *yin* für das Intervall A-dis aufgenommen mit der E-Tonabnehmerspule

werden. In diesem Fall wurde mit einem Abschlag erst die E- und dann die A-Saite angeschlagen. Die Aperiodizität für den Ton e ist geringer als für den Ton fis. Das kann daran liegen, dass beim Anschlag von e der Ton H schon an Leistung und Amplitude verloren hat und der lautere Ton e dominanter klingt.

Verschiedene Intervalle hintereinander: Es werden folgende Intervalle in Vierteln und Achteln bei 120 BPM gespielt: d+a, c+g, Ais+f, A+e. Der erstgenannte Ton wird auf der E- und der zweitgenannte Ton auf der A-Saite gespielt. In Abbildung 8.23 auf Seite 121 ist für die E-Tonabnehmerspule zu erkennen, dass der Graph von f_0 kaum Schwankungen um die erwarteten Frequenzen aufweist und die Tonlängen und -pausen korrekt erkannt werden. Die Aperiodizität ist über die gesamte Aufnahme sehr gering, abgesehen von den Tonwechseln. Für die Abbildung 8.24 auf Seite 122 wurde für die Aufnahme der A-Tonabnehmerspule der „plotthreshold“ wieder auf 0,5 erhöht, da die

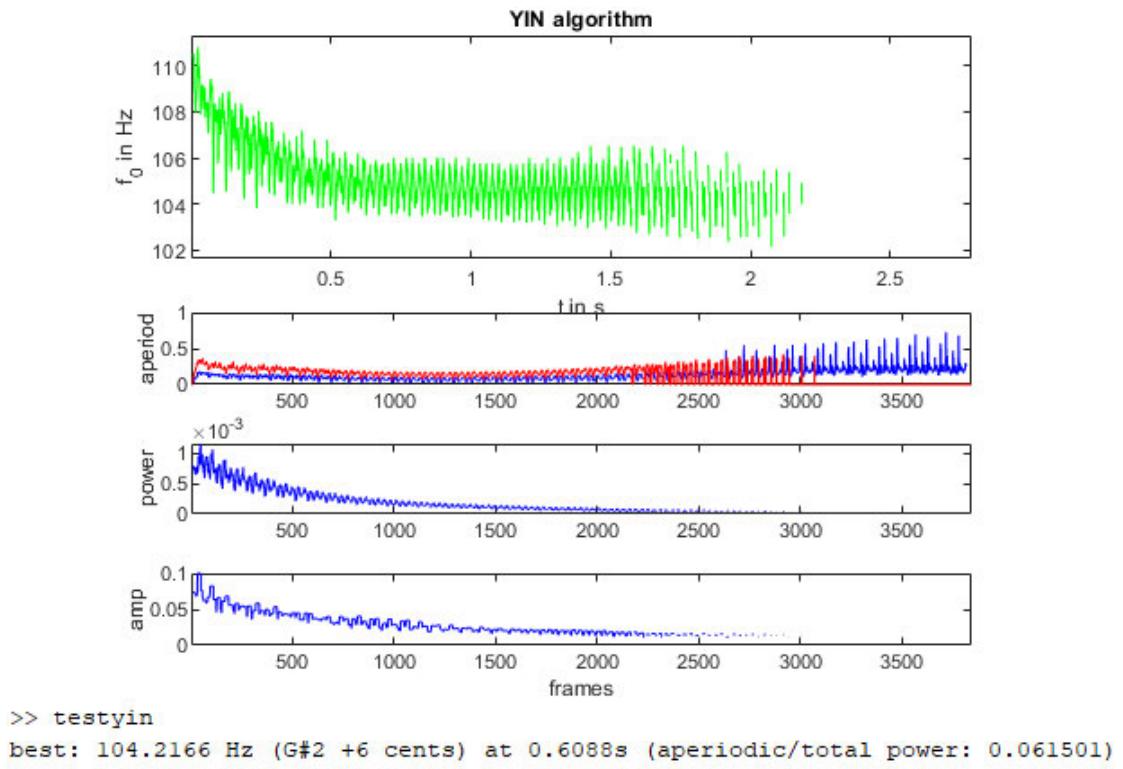


Abbildung 8.18: Ausgabe der Funktion *yin* für das Intervall A-dis aufgenommen mit der A-Tonabnehmerspule

Aperiodizität deutlich über dem vorherigen Schwellwert von 0,15 liegt. Der Graph von f_0 weist Schwankungen um den Bereich der erwarteten Frequenzen auf, aber die einzelnen Töne werden korrekt erkannt. Auch die Tonlängen und -pausen stimmen, auch wenn bei den Flanken der Tonwechsel teilweise Frequenzsprünge auftreten.

Polyphone Melodien auf allen Saiten: Zuletzt wird ein Sample mit gebrochenen Akkorden und Basstönen gespielt, siehe Abbildung 8.25 auf Seite 123. Die Töne A, H und c werden auf der A-Saite gespielt, der Ton E wird auf der E-Saite gespielt. Für die A-Saite wird zusätzlich der Analyse-Parameter min. f_0 auf 110 Hz erhöht. In den Abbildungen 8.26 und 8.27 auf der Seite 124 sind die Analyse-Ergebnisse für die E- und A-Saite dargestellt. Für die E-Saite ist zu erkennen, dass nur der Ton E am Ende des Samples erkannt wurde. Außerdem wurde der Anschlag des Ton A im vorletzten Takt erfasst, aber mit einer geringen Signalleistung als in der Aufnahme des Tonabnehmers

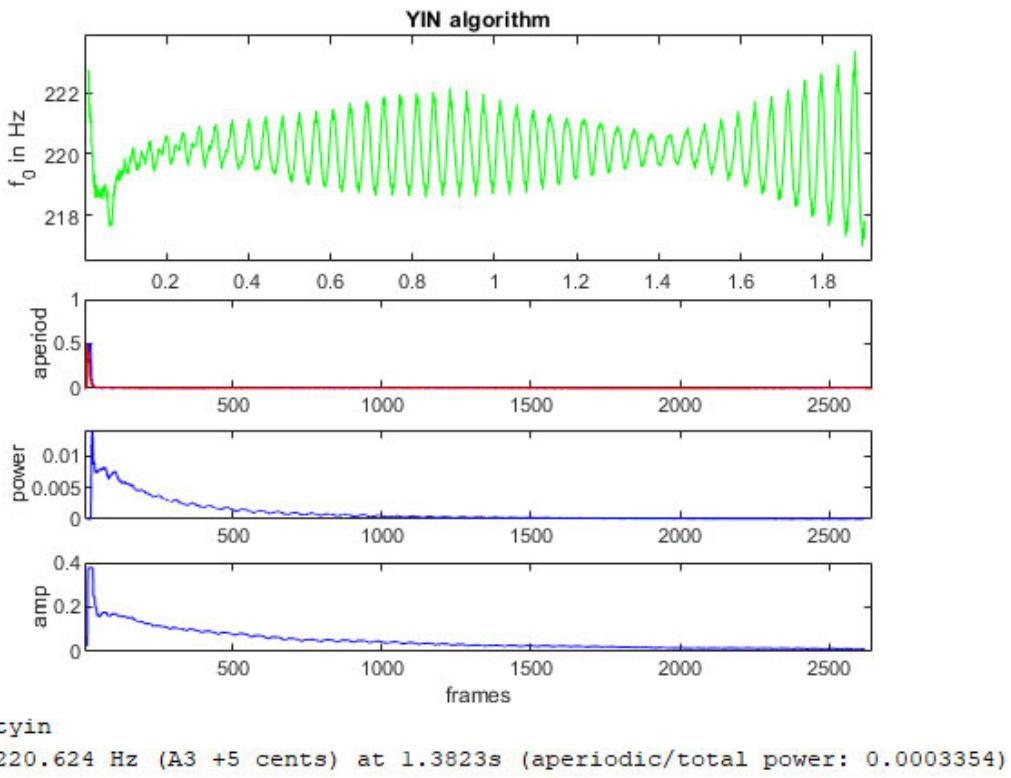


Abbildung 8.19: Ausgabe der Funktion *yin* für das Intervall a-g aufgenommen mit der E-Tonabnehmerspule

der A-Saite. Für die A-Saite wurden alle A- und c-Töne erfasst. Lediglich bei den Tönen H im sechsten und achten Takt schwankt die Aperiodizität stark, sodass dort nur einzelne Frequenzwerte vorliegen. Auch wenn es in beiden Abbildungen schwer zu erkennen ist, sind, bis auf die zwei H-Töne auf der A-Saite, die Töne korrekt in Achtel-Noten und lange Noten aufgeteilt. Dies kann man an den Ausschlägen der Aperiodizität erkennen.

8.2.9 Zusammenfassung der Analyse-Ergebnisse

Monophone Melodien, sowohl auf nur einer als auch auf allen Gitarren-Saiten gespielt, werden größtenteils richtig erkannt in Bezug auf Tonhöhe und Tonlänge. Teilweise können Oktavfehler auftreten. Bei Tonwechseln muss der Parameter „plotthreshold“ so justiert werden, dass steile Frequenzflanken unterdrückt werden, aber auch die Tonlänge nicht signifikant zu kurz ausfällt. Über den neu eingeführten Schwellwert für die Leistung

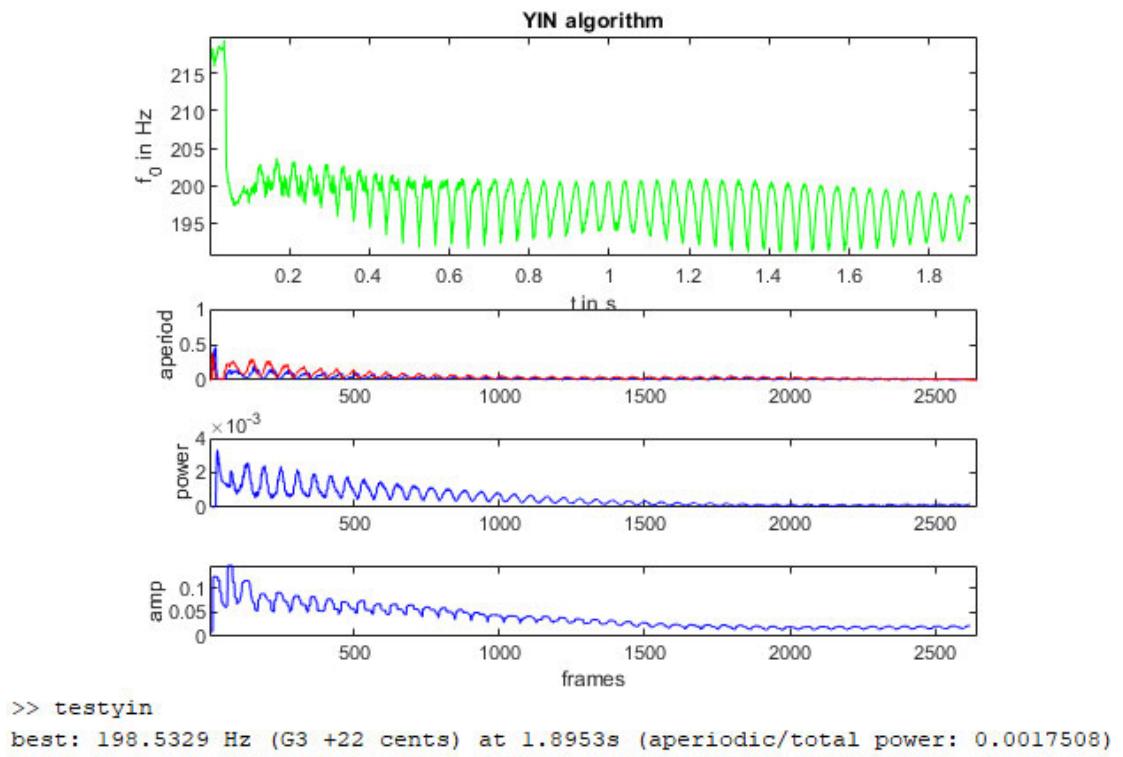


Abbildung 8.20: Ausgabe der Funktion *yin* für das Intervall a-g aufgenommen mit der A-Tonabnehmerspule

wird die Analyse von zu leisen Tönen unterdrückt, sodass Frequenzschätzungen für stille Abschnitte ignoriert werden. Ist dieser Schwellwert aber zu hoch, können Töne beim Ausklingen unterbrochen oder gekürzt werden. Die Tonabnehmerspulen nehmen auch gespielte Töne der Nachbarsaiten auf, wobei dies mit geringerer Signalleistung erfolgt. Dementsprechend muss nach der Analyse noch bestimmt werden, welcher Ton auf welcher Saite tatsächlich gespielt wurde. Polyphone Passagen werden teilweise falsch oder gar nicht erkannt. Werden Intervalle oder Akkorde gleichzeitig und gleich lang gespielt, wird tendenziell der tiefere Ton korrekt erkannt. Für den höheren Ton muss meistens der „plotthreshold“ erhöht werden, da die Interferenz mit dem anderen Ton die Aperiodizität steigen lässt. Sind Intervalle und Akkorde aber mit Rhythmen kombiniert oder werden die involvierten Töne nicht gleich lang gespielt, werden sie eher korrekt zugeordnet. Generell ist für die Analyse auf eine saubere und präzise Spieltechnik zu achten, damit unsaubere Handbewegungen nicht fälschlicherweise als Ton interpretiert werden.

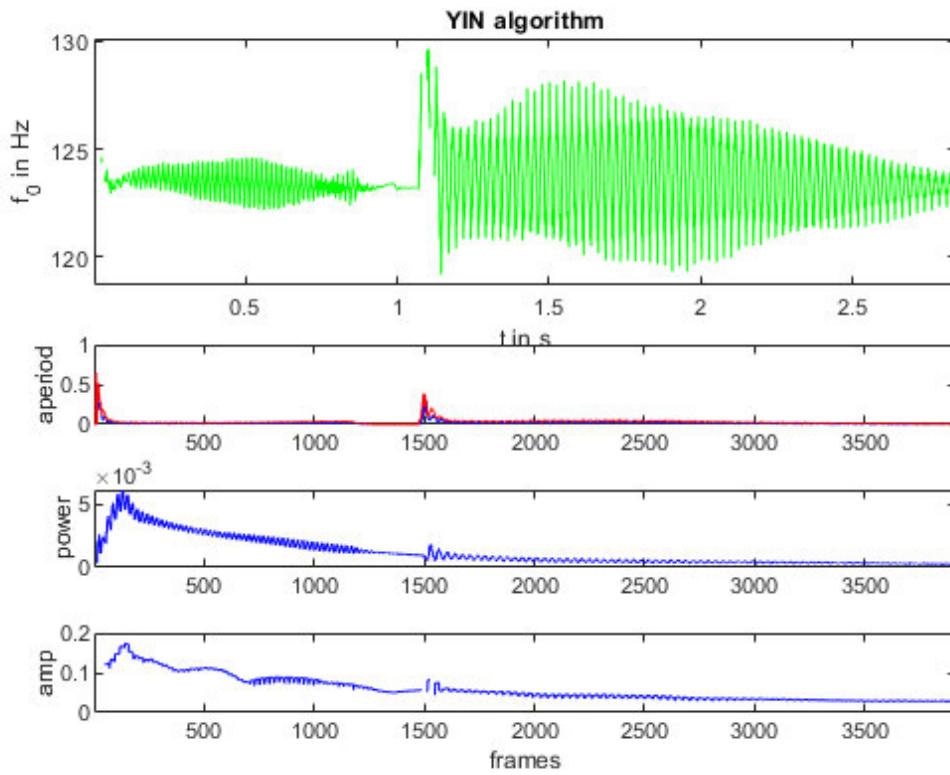


Abbildung 8.21: Ausgabe der Funktion *yin* für das Intervall H-fis/e aufgenommen mit der E-Tonabnehmerspule

8.3 MIDI-Konvertierung

Zur Konvertierung der Ergebnisse der *yin*-Funktion in eine MIDI-Datei müssen als erstes mehrfach aufgenommene Töne bearbeitet werden, sodass jeder Ton nur einer Saite zur Zeit zugeordnet wird. Im Anschluss müssen aus den Frequenzmessungen die Tonlängen und maximalen Tonamplituden ermittelt werden. Zuletzt werden die Zeitangaben in VLA umgerechnet und alle Informationen samt Datei- und Track-Headern in eine MIDI-Datei geschrieben.

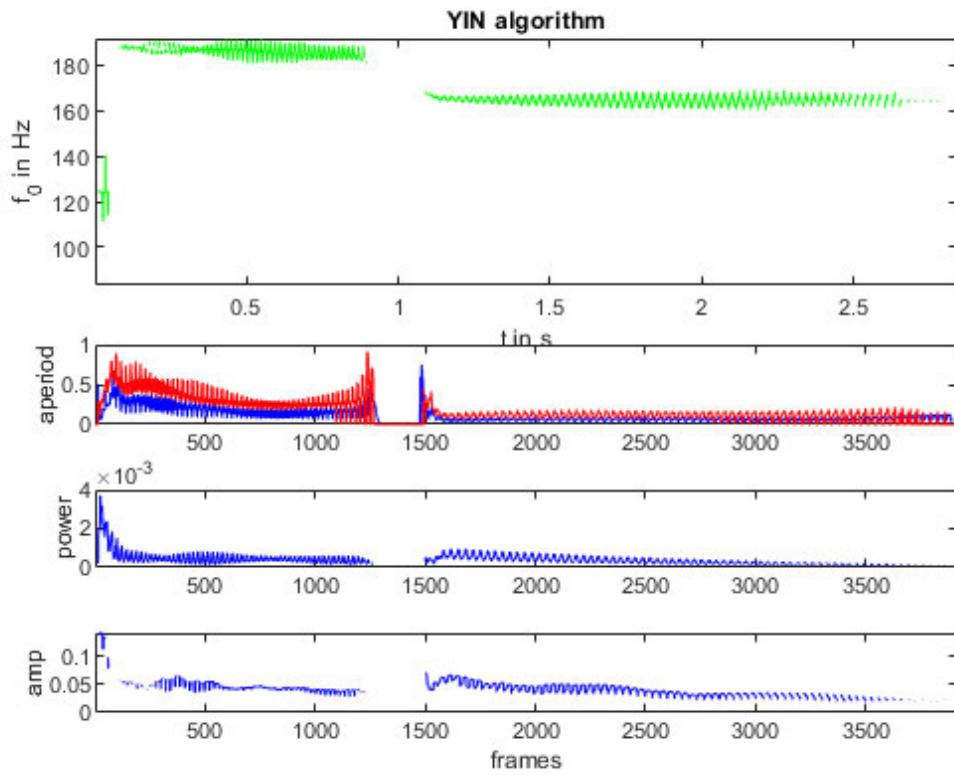


Abbildung 8.22: Ausgabe der Funktion *yin* für das Intervall H-fis/e aufgenommen mit der A-Tonabnehmerspule

8.3.1 Zuordnung der ermittelten Frequenzen zu den Tonspuren

Wie in Kapitel 8.2.7 gezeigt, nehmen Tonabnehmerspulen Töne benachbarter Saiten mit geringerer Amplitude auf. Diese Töne müssen in den Frequenzmessungen der jeweiligen Saite gelöscht werden, da der Ton von der Nachbarsaite stammt. Dazu werden in der Funktion *YINMIDI* alle Frequenzmessungen durchgegangen und wenn zu einem Zeitpunkt ein Ton auf mehreren Saiten ermittelt wurde, wird der Wert mit der höchsten Signalleistung behalten und die restlichen Werte für die anderen Saiten auf 0 gesetzt. Um die Töne leichter vergleichen zu können, werden die ermittelten Frequenzen über die folgende Gleichung in die korrespondierenden MIDI-Tonnummern umgerechnet [54]:

$$m = 12 \cdot \log_2 \left(\frac{f_m}{440 \text{ Hz}} \right) + 69 \quad (8.1)$$

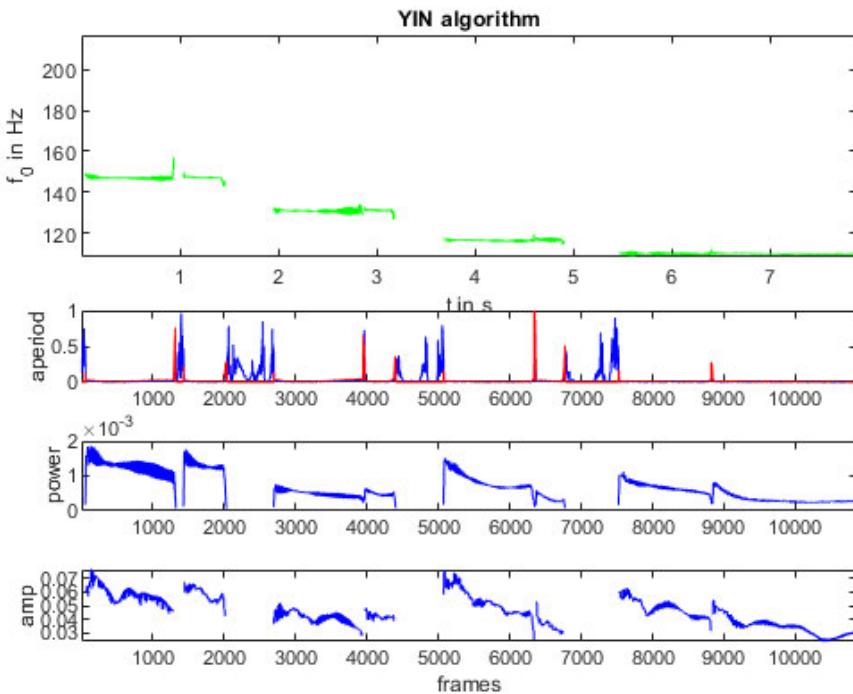


Abbildung 8.23: Ausgabe der Funktion *yin* für verschiedene Intervalle aufgenommen mit der E-Tonabnehmerspule

8.3.2 Konvertierung in MIDI-Angaben

Nachdem doppelt aufgenommene Töne korrigiert wurden, können deren Tonlängen, Pausen und Amplituden ermittelt werden. Dies erfolgt in der Funktion *midiConverter*. Sie benötigt die Ergebnis-Struktur der Funktion *yin*, eine Referenzamplitude für die Tonamplituden und einen BPM- und Ticks-Wert für Zeitangaben. Zuerst werden die ermittelten Tonamplituden mithilfe der Referenzamplitude in dB auf den Wertebereich 0-127 skaliert. Amplitudenwerte, die rechnerisch über dem Wert 127 liegen, werden auf diesen begrenzt. Anschließend werden die Frequenzmessungen und Tonamplituden iteriert und dabei gezählt, wie oft ein Ton hintereinander auftritt und welche Amplitude die höchste in diesem Bereich ist. Daraus wird dann die Information für einen MIDI-Ton erstellt. Bei Pausen und Unterbrechungen wird der vorherige MIDI-Ton als Tonnummer verwendet. Zuletzt wird die Anzahl an Messwerten für einen MIDI-Ton in die korrespondierende Anzahl MIDI-Ticks umgerechnet. Dazu wird erst die Tonlänge in s errechnet. Dafür wer-

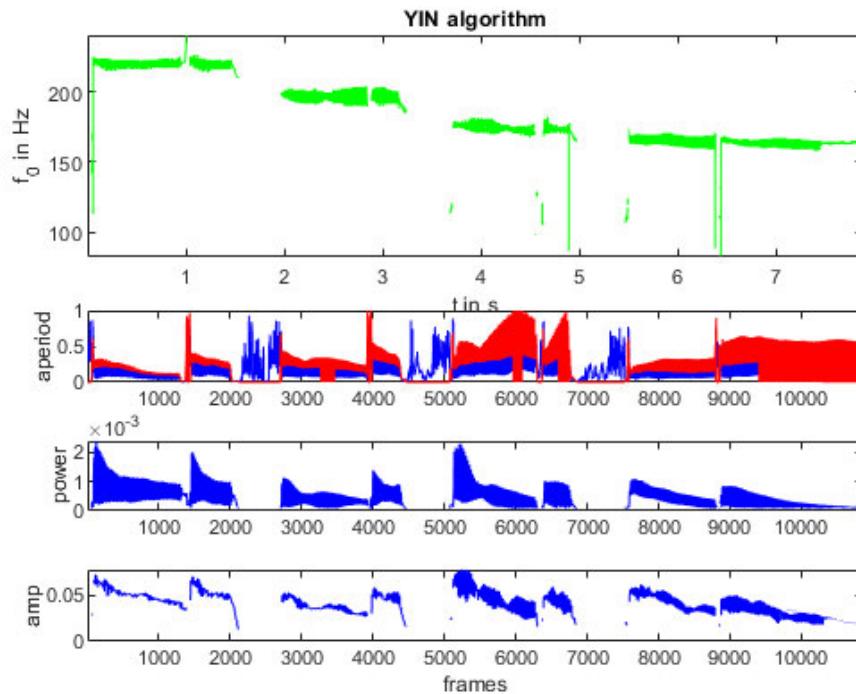


Abbildung 8.24: Ausgabe der Funktion *yin* für verschiedene Intervalle aufgenommen mit der A-Tonabnehmerspule

den die Angabe zur Abtastrate und die Schrittweite der Analyse in *yin* benötigt. Ein Messwert kann der Zeitspanne $\frac{\text{Schrittweite}}{\text{Abtastrate}}$ zugeordnet werden. Als Nächstes werden die übergebenen Informationen zu den gewünschten BPM und Anzahl an MIDI-Ticks pro Viertelnote benötigt. Mithilfe der BPM wird angegeben, wie viele Viertelnoten pro Minute erklingen sollen. Daraus lässt sich ableiten wie lang eine Viertelnote ist. Dann lässt sich bestimmen, aus wie vielen Viertelnoten die Tonlänge besteht. Zuletzt kann mit der Anzahl an n Ticks pro Viertelnote die Tonlänge in Ticks ausgedrückt werden [48]:

$$\text{Tonlänge [s]} = N_{\text{Messwerte}} \cdot \frac{\text{Schrittweite}}{\text{Abtastrate [Hz]}} \quad (8.2)$$

$$\text{Tonlänge [Ticks]} = \text{Tonlänge [s]} \cdot \text{BPM} \left[\frac{\text{Viertelnoten}}{60 \text{ s}} \right] \cdot n \left[\frac{\text{Ticks}}{\text{Viertelnote}} \right] \quad (8.3)$$

Der Inhalt des Rückgabe-Array der Funktion *midiconverter* ist in Tabelle 8.1 auf Seite 123 beispielhaft dargestellt.



Abbildung 8.25: Polyphone Melodie mit gebrochenen Akkorden und Basstönen

Tabelle 8.1: Beispielausgabe der Funktion *midiConverter*

Index	Tonnummer	Lautstärke	Tonlänge
1	40	50	480
2	40	0	480
3	45	50	960
4	45	0	0

8.3.3 Übersetzung in Bytes und Datei-Erstellung

Darstellung der Tonlängen als VLA

Die Werte für die Tonlängen müssen in VLA konvertiert werden, bevor sie in die MIDI-Datei geschrieben werden können. Dazu wird die Funktion *writeVarLen* benutzt, die sich am C-Beispielcode aus [48] orientiert, siehe Abbildung A.5. Die Tonnummern, Lautstärken und Tonlängen werden nun als Cell-Array zusammengefasst, da die VLA zwischen einem und vier Bytes umfassen können und damit die Array-Elemente hierfür verschieden groß sind. Das korrespondierende Cell-Array für die Werte in Tabelle 8.1 ist in Tabelle 8.2 dargestellt.

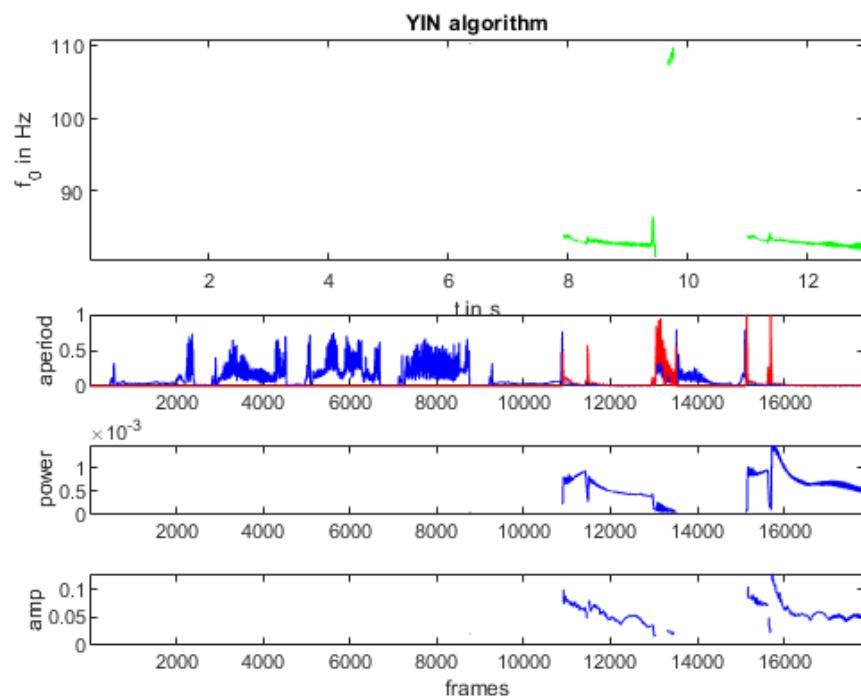


Abbildung 8.26: Ausgabe der Funktion *yin* für die polyphone Melodie aufgenommen mit der E-Tonabnehmerspule

Tabelle 8.2: Cell-Array mit Tonnummern, Lautstärken und Tonlängen als VLA

Index	Tonnummer	Lautstärke	Tonlänge
1	40	50	[131, 96]
2	40	0	[131, 96]
3	45	50	[135, 64]
4	45	0	0

Datei-Erstellung

Zum Erstellen der MIDI-Datei wird die Funktion *midiFile* verwendet. Sie erstellt einen Dateikopf und bis zu sechs MIDI-Tracks für das Dateiformat 1, siehe Kapitel 2.12.3.

Funktionsparameter:

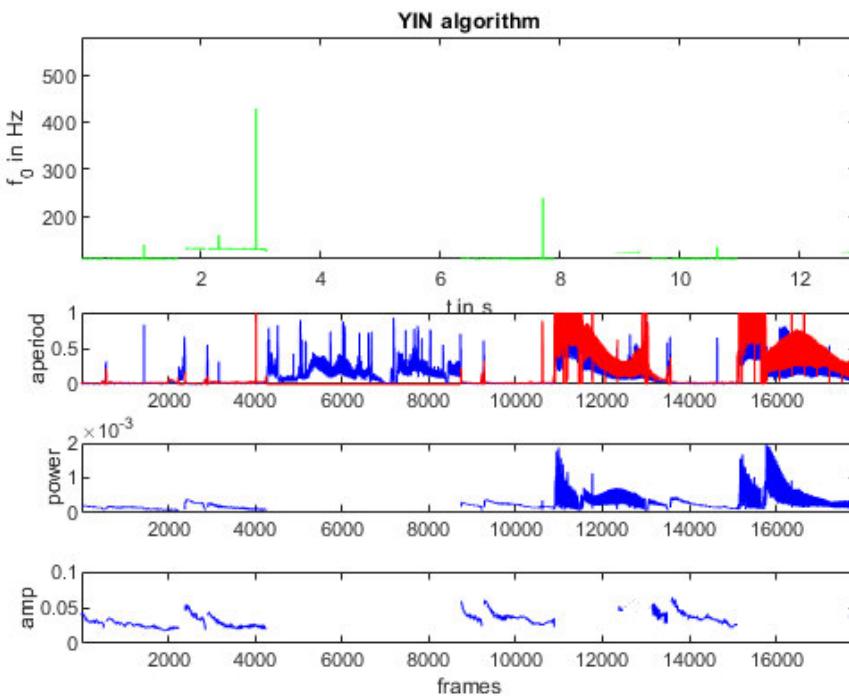


Abbildung 8.27: Ausgabe der Funktion *yin* für die polyphone Melodie aufgenommen mit der A-Tonabnehmerspule

- fileName: Dateiname der MIDI-Datei, die erstellt oder ergänzt werden soll
- partOfFile: gibt an, ob eine neue Datei samt Datei-Header erstellt oder ein Track an eine bestehende Datei angefügt werden soll
- midiTracks/midiArray: wenn partOfFile=1, enthält midiTracks die Anzahl an folgenden MIDI-Tracks, die für die Formatangabe im Datei-Header benötigt wird. Wenn partOfFile=0, enthält midiArray ein Array wie in Tabelle 8.1 dargestellt.
- p: eine Struktur mit optionalen Angaben zu BPM, MIDI-Ticks und den Track-Namen.

Datei-Header: Zuerst wird die Datei mit dem Namen „fileName“ mit der Funktion *fopen* erstellt und zum Schreiben geöffnet. Falls eine Datei mit dem Namen bereits existiert, wird ihr Inhalt gelöscht und die Datei neu beschrieben. Anschließend wird der

Datei-Header mit der Funktion *fwrite* in die Datei geschrieben. *fwrite* akzeptiert entweder Zeichen-Arrays, numerische Arrays oder auch String-Arrays. Zeichen-Arrays werden ASCII-kodiert in die Datei geschrieben, numerische Werte werden unverändert übernommen. Die Anzahl von MIDI-Ticks pro Viertelnote muss vorab in zwei Bytes konvertiert werden. Dafür wird die Hilfsfunktion *writeBytes* verwendet, die ähnlich wie *writeVLA* arbeitet, siehe Abbildung A.6. Anschließend wird die Datei mit *fclose* wieder geschlossen.

MIDI-Track: Für die MIDI-Tracks wird die Datei „fileName“ geöffnet und am Ende der Datei weitergeschrieben. Falls keine Parameter für den Track-Header übergeben werden, wird von 120 BPM in einem 4/4-Takt ausgegangen. Die Angabe zu den BPM wird mit der Funktion *writeBytes* in drei Bytes konvertiert. Anschließend wird das „midiArray“ wie vorab erwähnt in ein Cell-Array konvertiert und dabei gezählt, aus wie vielen Bytes es besteht. Falls der Track mit einer Pause beginnt, werden zwei Bytes von der Gesamtzahl abgezogen, da zu Beginn keine Tonnummer und Lautstärke für eine Delta-Zeit benötigt werden. Anschließend werden noch 14 Bytes für den Track-Header, ein Byte für die Voice-Nachricht „Note On“ und drei Bytes für das Event „Track-Ende“ dazu gezählt. Falls ein Track-Name übergeben wurde, kann dieser als char-Array direkt in die Datei geschrieben werden und seine Anzahl an chars wird der Gesamtzahl an Bytes hinzugefügt. Die Gesamtzahl an Bytes muss dann noch um ein Byte erhöht werden, da sonst ein Kodierungsfehler beim Öffnen der MIDI-Datei im MIDI-Editor auftritt. Die Gesamtzahl der genutzten Bytes wird mit *writeBytes* in vier Bytes konvertiert. Nach der Erstellung des Track-Headers wird dann das Cell-Array iteriert und die Elemente nacheinander in die Datei geschrieben. Der Track wird dann mit dem Event „Track-Ende“ beendet und die Datei geschlossen.

8.3.4 Öffnen der erstellten MIDI-Datei im MIDI-Editor

A auf der E-Saite: Es wird die Datei „E+A-EA2-solo.wav“, siehe Abbildung A.8, analysiert und die Ergebnisse für die E- und A-Saite inklusive Reduktion doppelter Töne in die Datei „EA2-solo.mid“ konvertiert. Für die A-Saite wird zusätzlich der Analyse-Parameter $\min. f_0$ auf 110 Hz erhöht. Für die Konvertierung werden 480 Ticks pro Viertelnote, 120 BPM, -12 dB als Referenzamplitude und ‚Gitarre‘ als Track-Name als Parameter verwendet. Die MIDI-Datei kann ohne Probleme geöffnet werden. Der Track für die E-Saite wird blau und der Track für die A-Saite rot dargestellt. Links im Editor

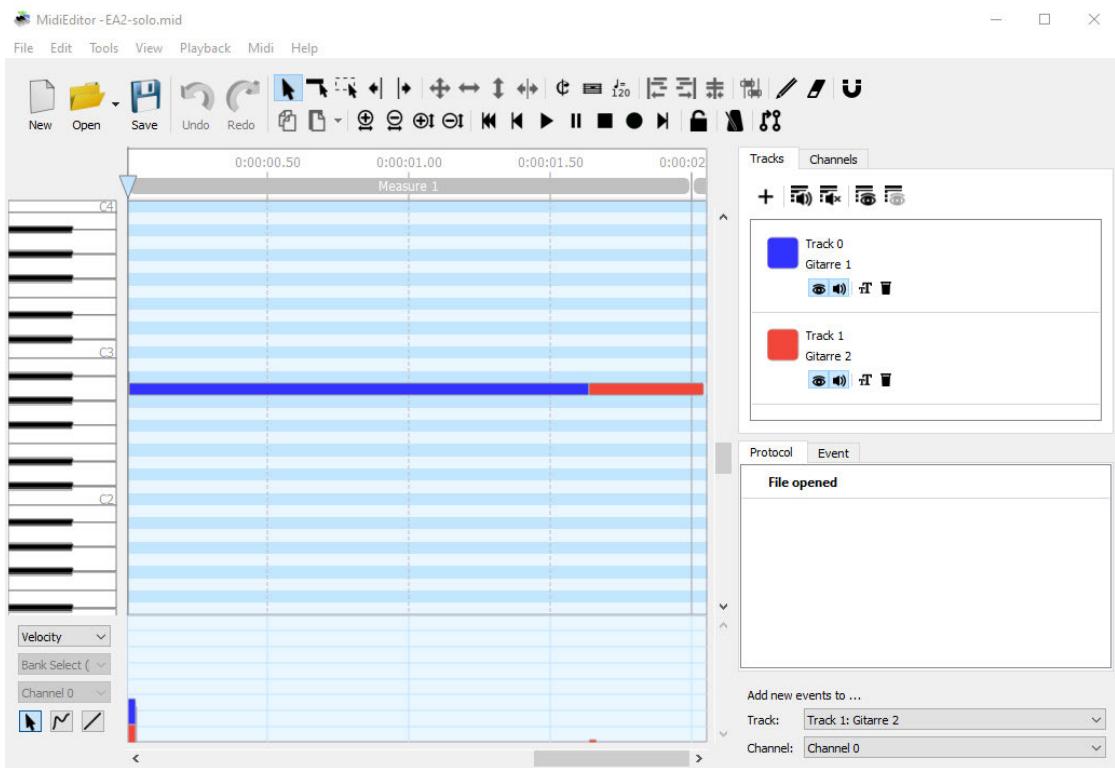


Abbildung 8.28: Geöffnete Datei *EA2-solo.mid* im MIDI-Editor

wird zur tonalen Orientierung eine Klaviatur eingeblendet. Am Anfang der Datei, in der Abbildung 8.28 nicht zu erkennen, wird für zwei Ticks der Ton Ais statt A erzeugt. Am Ende der Tracks ist zu erkennen, dass der Ton A auch der A-Saite zugeordnet wurde, obwohl er auf der E-Saite gespielt wurde. Blendet man den Track 0 der E-Saite aus, ist zu erkennen, dass der Großteil des Tracks 1 der A-Saite keine Töne enthält, so wie es idealerweise sein sollte, siehe Abbildung 8.29. Klickt man das letzte rote Event an, kann man in den Event-Details sehen, dass der Ton A auf der A-Saite nur mit der Lautstärke 4 von 127 erfasst wurde, was kaum lauter als Stille ist. Das heißt akustisch klingt die Datei fehlerfrei, inhaltlich müssen nur wenige Events nachträglich bearbeitet oder gelöscht werden, um das Gespielte korrekt darzustellen.

Polyphone Melodie: Es wird die Datei „E+A-polyphoneMelodie.wav“, siehe Abbildung 8.26 und Abbildung 8.27, analysiert und die Ergebnisse für die E- und A-Saite inklusive Reduktion doppelter Töne in die Datei „polyphoneMelodie.mid“ konvertiert.

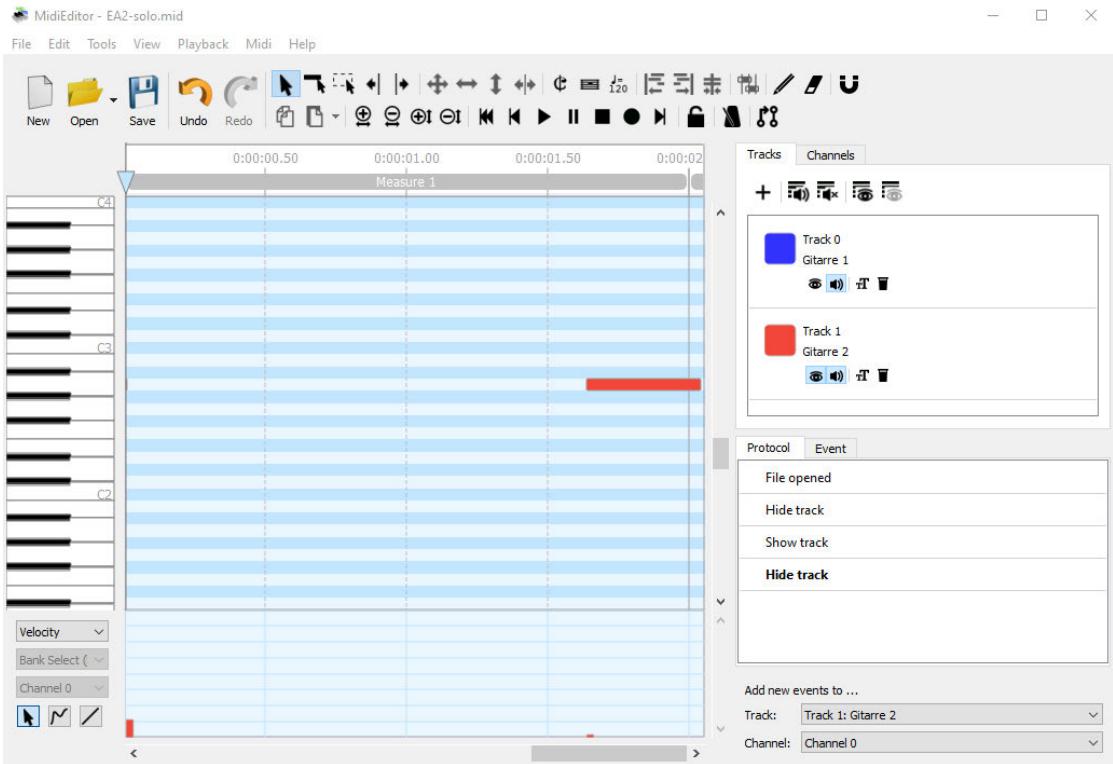


Abbildung 8.29: Geöffnete Datei *EA2-solo.mid* im MIDI-Editor, Track 0 ausgeblendet

Für die Konvertierung werden 480 Ticks pro Viertelnote, 115 BPM, -18 dB als Referenzamplitude und ‚Gitarre‘ als Track-Name als Parameter verwendet. Da die Noten in 6/8 notiert sind, wird nachträglich in der MIDI-Datei ein Meta-Event für die entsprechende Zeitsignatur ergänzt. Aktuell bietet die Funktion *midiFile* noch keine Option Parameter für die Zeitsignatur zu übergeben. Für die E-Saite wird als minimale f_0 80 Hz und für die A-Saite 110 Hz übergeben. Die erstellte MIDI-Datei ist in Abbildung 8.31 zu sehen. Für den blauen Track der E-Saite ist zu erkennen, dass es kaum Artefakte durch falsche MIDI-Events gibt. Lediglich bei der Tonlänge gibt es teilweise mehrere Unterbrechungen die wenige Ticks lang sind, aber beim Hören wahrnehmbar sind. Für die A-Saite gibt es am Ende des sechsten Taktes viele kurze Artefakte für den Ton H, der in dem Takt aber schon zur Taktmitte gespielt wurde und hier nur als Nachhall erfasst wird. Außerdem sind für den Schlussakkord im achten Takt wieder für die A-Saite Artefakte für den Ton H zu sehen, obwohl der Ton schon früher angeschlagen wurde.

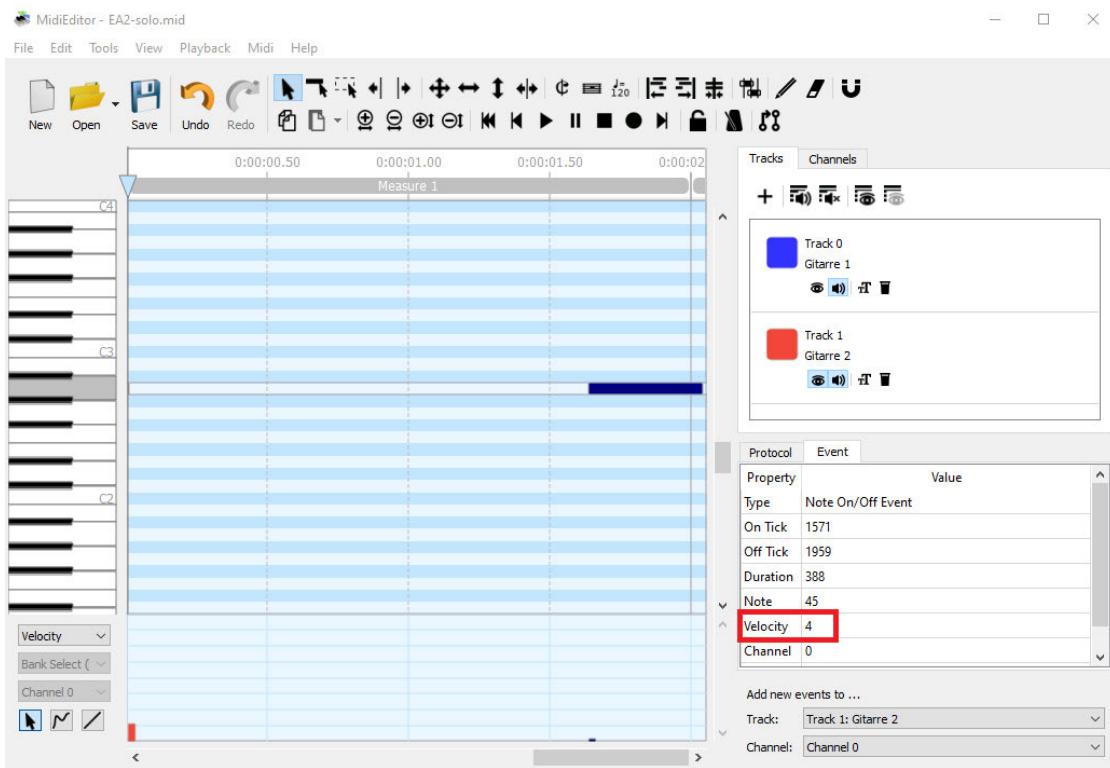


Abbildung 8.30: Geöffnete Datei *EA2-solo.mid* im MIDI-Editor, letztes Event markiert

8.4 Latenz

Zur Bestimmung der Latenz der Software wird erst die Latenz der Funktion *yin* alleine betrachtet und anschließend die Latenz von YIN-Analyse und MIDI-Konvertierung zusammen. Dazu wird die integrierte Programmlaufzeitmessung von GNU Octave 9.1.0 auf einem Computer mit einem 3,6 GHz AMD Ryzen 5 Prozessor und einem 16 GB DDR4-3200 RAM ausgeführt.

8.4.1 YIN

Die Sub-Funktion *yin_helper* benötigt für einen erwarteten Frequenzbereich von 80 Hz bis 1174 Hz bei einer Schrittweite von 32 Samples mindestens 1152 Datensample um mindestens einmal ausgeführt zu werden. Als Ergebnis liegen dann 36 Frequenzschätzungen

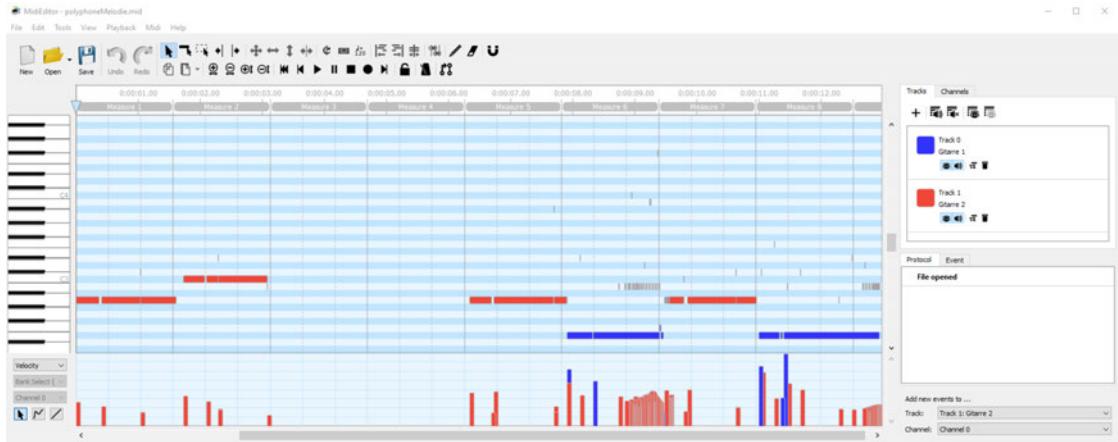


Abbildung 8.31: Geöffnete Datei „polyphoneMelodie.mid“ im MIDI-Editor

vor. Zum Testen wird die Datei „E+a-polyphoneMelodie.wav“ mittels *audioread* als Variable eingelesen und samt der Angabe zur Abtastfrequenz von 44,1 kHz als Workspace abgespeichert. Dieser Workspace wird immer vor der Programmlaufzeitmessung eingelesen, da der Aufruf von *audioread* ansonsten jedes Mal Zeit benötigt, die die Programmlaufzeitmessung von *yin* verfälscht. Anschließend werden in der Datei *testyin* die ersten 1152 Sample des ersten Datenspalte und die Analyseparameter an *yin* übergeben und ein Rückgabeparameter angegeben, um die Ausgabe des Ergebnis-Plots zu vermeiden. Die gemessene Programmlaufzeit beträgt 2,63 ms und teilt sich gemäß Abbildung 8.32 auf Seite 132 auf 47 genutzte Funktionen und Befehle auf. Die Laufzeit der zwei Aufrufe von *yin_helper* beträgt summiert 0,21 ms.

Unter realen Bedingungen würden die 1152 benötigten Datensamples nicht sofort zur Verfügung stehen sondern müssten erst mit der Abtastrate von 44,1 kHz aufgenommen werden. Das heißt die Aufnahme der Samples dauert mindestens $1152/44100 \text{ Hz} = 26,1 \text{ ms}$. Insofern muss erst die Anzahl minimal benötigter Samples für *yin_helper* reduziert werden, um die gesamte Latenz zu reduzieren.

Wiederholt man die Programmlaufzeitmessung für den gesamten Dateiinhalt (571.747 Datensamples in einer Tonspur), beträgt die Programmlaufzeit 3,09 s. In Abbildung 8.33 auf den Seite 133 ist zu erkennen, dass die zeitintensivsten Funktionen die Operationen des YIN-Algorithmus sind. Die aufgeführten Zeiten in der Spalte „Time“ sind als Summe der Laufzeiten aller Funktionsaufrufe in der Spalte „Calls“ zu verstehen.

8.4.2 YIN und MIDI-Konvertierung

Als Nächstes wird die Funktion *YINMIDI* getestet. Zunächst erhält auch sie 1152 Datensamples für beide Tonspuren, die vorab in den Workspace geladen wurden. Die gemessene Programmlaufzeit beträgt 14,28 ms und teilt sich gemäß Abbildung 8.34 auf Seite 134 auf 87 genutzte Funktionen und Befehle auf, wobei nur die 50 zeitintensivsten dargestellt sind. Die Laufzeit aller Aufrufe von *yin_helper* beträgt diesmal 2,1 ms und von *midiFile* 1,16 ms. Die Programmlaufzeitmessung wird für den gesamten Dateiinhalt (571.747 Datensamples jeweils in zwei Tonspuren) wiederholt. Die Programmlaufzeit beträgt 7,68 s. In Abbildung 8.35 auf Seite 135 ist zu erkennen, dass die zeitintensivsten Funktionen die Operationen des YIN-Algorithmus und die Funktion *unique*, die für die Reduktion doppelt aufgenommener Töne benutzt wird, sind. Die aufgeführten Zeiten in der Spalte „Time“ sind als Summe der Laufzeiten aller Funktionsaufrufe in der Spalte „Calls“ zu verstehen.

>> profshow(result, 47)	#	Function Attr	Time (s)	Time (%)	Calls
26 yink>yin_helper	0.001		28.55		2
2 yin	0.000		16.83		1
18 yink	0.000		12.79		1
1 testyin	0.000		7.88		1
33 rdiff_inplace	0.000		5.82		1
5 sf_info R	0.000		5.49		2
25 sf_wave	0.000		5.37		2
46 clear	0.000		2.03		1
36 minparabolic	0.000		1.88		1
23 zeros	0.000		1.68		16
27 rsmooth	0.000		1.28		2
9 sf_format	0.000		1.00		1
37 cumnorm_inplace	0.000		0.93		1
35 postfix '	0.000		0.73		2
7 isa	0.000		0.64		6
8 isfield	0.000		0.64		21
20 binary +	0.000		0.48		37
15 binary /	0.000		0.46		24
43 sf_cleanup	0.000		0.46		1
11 size	0.000		0.45		6
34 rsum_inplace	0.000		0.45		4
22 NaN	0.000		0.38		11
21 round	0.000		0.33		8
31 max	0.000		0.30		3
38 dftoperiod2	0.000		0.30		1
6 prefix !	0.000		0.26		34
12 binary *	0.000		0.26		14
24 min	0.000		0.23		3
39 dftoperiod	0.000		0.22		1
3 nargin	0.000		0.19		12
4 binary <	0.000		0.19		16
40 interp_inplace	0.000		0.19		1
19 binary -	0.000		0.18		17
41 binary .^	0.000		0.15		3
10 strcmp	0.000		0.14		4
13 isempty	0.000		0.14		6
42 eps	0.000		0.12		2
45 isnan	0.000		0.12		3
16 ceil	0.000		0.11		7
30 abs	0.000		0.08		1
29 floor	0.000		0.07		6
44 binary ./	0.000		0.07		1
14 binary >	0.000		0.04		8
17 mod	0.000		0.04		1
28 end	0.000		0.04		2
32 binary ==	0.000		0.00		1
47 nargout	0.000		0.00		1

Abbildung 8.32: Detaillierte Aufteilung der Programmlaufzeit der YIN-Analyse für 1.152 Datensamples

>> profshow(result, 49)	#	Function Attr	Time (s)	Time (%)	Calls
26 yink>yin_helper	2.073		67.07		66
33 rdiff_inplace	0.164		5.30		65
34 rsum_inplace	0.118		3.82		53561
36 minparabolic	0.090		2.91		65
41 dftoperiod	0.079		2.56		17832
38 dftoperiod2	0.079		2.55		17832
20 binary +	0.076		2.46		250759
42 interp_inplace	0.060		1.94		17832
35 postfix '	0.050		1.61		17897
31 max	0.049		1.58		35665
19 binary -	0.038		1.24		89620
43 binary .^	0.034		1.10		53496
44 eps	0.030		0.96		35664
37 cumnorm_inplace	0.026		0.84		65
24 min	0.022		0.72		17898
18 yink	0.019		0.62		1
23 zeros	0.019		0.61		400
30 abs	0.014		0.45		17832
12 binary *	0.011		0.34		89362
16 ceil	0.009		0.29		35797
15 binary /	0.009		0.28		72052
2 yin	0.005		0.17		1
29 floor	0.005		0.15		18029
25 sf_wave	0.004		0.14		66
27 rsmooth	0.003		0.11		66
1 testyin	0.002		0.07		1
5 sf_info R	0.002		0.05		2
40 mininrange	0.001		0.02		65
21 round	0.000		0.01		392
11 size	0.000		0.01		134
46 binary ./	0.000		0.00		1
8 isfield	0.000		0.00		149
39 ones	0.000		0.00		65
6 prefix !	0.000		0.00		290
10 strcmp	0.000		0.00		68
4 binary <	0.000		0.00		400
48 clear	0.000		0.00		1
3 nargin	0.000		0.00		204
13 isempty	0.000		0.00		134
28 end	0.000		0.00		66
14 binary >	0.000		0.00		200
9 sf_format	0.000		0.00		1
47 isnan	0.000		0.00		3
7 isa	0.000		0.00		6
45 sf_cleanup	0.000		0.00		1
32 binary ==	0.000		0.00		65
22 NaN	0.000		0.00		11
17 mod	0.000		0.00		1
49 nargout	0.000		0.00		1

Abbildung 8.33: Detaillierte Aufteilung der Programmalaufzeit der YIN-Analyse für 571.747 Datensamples

>> profshow(result,50)	#	Function Attr	Time (s)	Time (%)	Calls
3	YINMIDI		0.003	18.15	1
30	yink>yin_helper		0.002	15.25	4
59	unique		0.001	9.47	38
75	midiFile		0.001	8.18	3
84	midiConverter		0.001	5.29	2
12	yin		0.001	4.92	2
76	fopen		0.001	4.34	3
22	yink		0.001	3.68	2
56	nonzeros		0.000	3.22	36
37	rdiff_inplace		0.000	2.10	2
29	sf_wave		0.000	1.77	4
1	testyin		0.000	1.60	1
82	fclose		0.000	1.58	3
13	sf_info	R	0.000	1.48	4
87	writeVarLen		0.000	1.15	4
81	fwrite		0.000	1.14	41
72	validsetargs		0.000	1.01	1
27	zeros		0.000	0.98	69
77	writeBytes		0.000	0.97	5
71	setdiff		0.000	0.94	1
40	minparabolic		0.000	0.66	2
58	find		0.000	0.58	39
50	clear		0.000	0.53	2
38	rsum_inplace		0.000	0.52	35
61	false		0.000	0.50	76
24	binary +		0.000	0.49	335
79	bitand		0.000	0.45	18
62	true		0.000	0.44	76
31	rsmooth		0.000	0.38	4
9	size		0.000	0.36	42
39	postfix '		0.000	0.36	13
43	dftoperiod		0.000	0.34	11
41	cumnorm_inplace		0.000	0.33	2
42	dftoperiod2		0.000	0.33	11
8	prefix !		0.000	0.32	161
15	sf_format		0.000	0.30	2
11	isfield		0.000	0.29	57
35	max		0.000	0.28	27
80	bitshift		0.000	0.27	18
23	binary -		0.000	0.26	167
44	interp_inplace		0.000	0.26	11
5	nargin		0.000	0.24	153
14	isa		0.000	0.23	15
6	binary <		0.000	0.23	178
65	any		0.000	0.22	28
4	cell		0.000	0.21	12
45	binary .^		0.000	0.18	33
19	binary /		0.000	0.17	94
68	length		0.000	0.17	92
10	binary >		0.000	0.16	166

Abbildung 8.34: Detaillierte Aufteilung der Programmlaufzeit der YIN-Analyse und MIDI-Konvertierung für 1.152 Datensamples

>> profshow(result,50)	#	Function Attr	Time (s)	Time (%)	Calls
30 yink>yin_helper	4.105		53.45		130
3 YINMIDI	0.536		6.97		1
59 unique	0.509		6.62		17995
37 rdiff_inplace	0.280		3.65		128
84 midiConverter	0.234		3.05		2
38 rsum_inplace	0.227		2.96		107147
56 nonzeros	0.216		2.81		17867
45 dftperiod	0.156		2.03		35673
24 binary +	0.155		2.02		538375
42 dftperiod2	0.147		1.92		35673
40 minparabolic	0.117		1.53		128
46 interp_inplace	0.114		1.48		35673
35 max	0.096		1.25		71414
39 postfix '	0.085		1.10		35801
23 binary -	0.080		1.04		215533
47 binary .^	0.067		0.87		107019
48 eps	0.060		0.78		71346
28 min	0.044		0.58		35803
41 cumnorm_inplace	0.044		0.58		128
27 zeros	0.035		0.45		6173
58 find	0.034		0.44		18059
61 false	0.032		0.42		35990
62 true	0.030		0.40		35990
22 yink	0.029		0.38		2
34 abs	0.027		0.36		35673
17 binary *	0.023		0.29		178776
19 binary /	0.017		0.22		144128
20 ceil	0.017		0.22		71608
5 nargin	0.013		0.16		54651
64 isrow	0.011		0.14		18059
68 length	0.011		0.14		35817
33 floor	0.009		0.12		36061
29 sf_wave	0.009		0.11		130
57 issparse	0.009		0.11		35862
31 rsmooth	0.008		0.11		130
75 midiFile	0.008		0.10		3
72 validsetargs	0.008		0.10		64
6 binary <	0.007		0.10		54988
71 setdiff	0.007		0.09		64
10 binary >	0.007		0.08		55151
88 writeVarLen	0.006		0.08		190
65 any	0.006		0.07		5316
9 size	0.005		0.07		5456
60 isnumeric	0.005		0.07		18123
8 prefix !	0.005		0.07		24236
63 numel	0.005		0.06		17995
67 class	0.004		0.06		5188
83 binary !=	0.004		0.05		35739
36 binary ==	0.004		0.05		31477
81 fwrite	0.002		0.03		600

Abbildung 8.35: Detaillierte Aufteilung der Programmalaufzeit der YIN-Analyse und MIDI-Konvertierung für 571.747 Datensamples

9 Test

In diesem Kapitel wird überprüft, ob die entwickelte Verarbeitungskette vom hexaphonischen Tonabnehmer bis zur MIDI-Konvertierung die Anforderungen aus Kapitel 4 erfüllt. Für den Erfüllungsgrad gibt es die Optionen „Ja“, „Nein“, „Teilweise“ und „Ausstehend“. Anforderungen mit der Priorität 2 müssen mindestens teilweise erfüllt worden sein.

9.1 Tonabnehmer

Die Beurteilung der funktionalen Anforderungen an den Tonabnehmer erfolgt in Tabelle 9.1 und die Beurteilung der nicht-funktionalen Anforderungen erfolgt in Tabelle 9.2. Die Anforderungen T-1, T-3 und T-5 sind vollständig erfüllt. Die Anforderung T-2 ist teilweise erfüllt und die Anforderung T-4 ist nicht erfüllt.

T-2: Es ist davon auszugehen, dass der hexaphonische Tonabnehmer vollständig funktioniert hätte, wenn er nicht aus dem Schraubstock gefallen wäre. Auf den Bau eines zweiten Tonabnehmers wurde aus Zeitgründen verzichtet, da noch zwei funktionierende Spulen vorhanden sind, die zum Testen von polyphonen Melodien ausreichen.

T-4: Um zu vermeiden, dass die Anschlussleitungen vom Tonabnehmer zur Vorverarbeitung auf dem Korpus der E-Gitarre befestigt werden müssen und so beim Spielen stören, wurden sie in das rückseitige Elektronikfach verlegt, aus dem sie herausgeführt werden müssen. Es bestand die Option die Öffnung für die montierte Klinkenbuchse zu verwenden, das würde aber die Parallelnutzung des noch vorhandenen Steg-Tonabnehmers verhindern. Deswegen wurde eine Lochbohrung mit 1 cm in das Elektronikfach vorgenommen, um die Anschlussleitungen herausführen zu können.

Tabelle 9.1: Erfüllungsgrad der funktionalen Anforderungen an den Tonabnehmer

Nr.	Priorität	Anforderung	Erfüllt
T-1	2	Der Tonabnehmer muss eine hexaphonische Bauart für sechs Saiten haben.	Ja
T-2	2	Der hexaphonische Tonabnehmer muss Saitenschwingungen in einzelne elektrische Signale mit gleicher Frequenz und Amplitudenverlauf umwandeln können.	Teilweise

Tabelle 9.2: Erfüllungsgrad der nicht-funktionalen Anforderungen an den Tonabnehmer

Nr.	Priorität	Anforderung	Erfüllt
T-3	2	Der hexaphonische Tonabnehmer muss die Gehäuseform eines Single Coil-Tonabnehmers haben.	Ja
T-4	1	Es dürfen keine irreversiblen Maßnahmen zur Platzschaffung an der E-Gitarre durchgeführt werden	Nein
T-5	1	Die Montage beinhaltet nur wieder lösbar Kontaktwege.	Ja

9.2 Instrumentenkabel

Die Beurteilung der funktionalen Anforderungen an das Instrumentenkabel erfolgt in Tabelle 9.3 und die Beurteilung der nicht-funktionalen Anforderungen erfolgt in Tabelle 9.4. Die Anforderungen K-1 und K-3 sind vollständig erfüllt. Die Anforderung K-2 ist teilweise erfüllt.

K-2: Die verwendeten Stifteleistenkontakte führen dazu, dass die Anschlussleitungen an der d- und g-Saite anschlagen und die Saitenschwingung stark dämpfen. Die Leitungen müssen stark gebogen werden, um dies zu verhindern.

Tabelle 9.3: Erfüllungsgrad der funktionalen Anforderungen an das Instrumentenkabel

Nr.	Priorität	Anforderung	Erfüllt
K-1	2	Das Instrumentenkabel muss sechs elektrische Signale und sechs Masseverbindungen von der E-Gitarre zur Vorverarbeitung übertragen können.	Ja

Tabelle 9.4: Erfüllungsgrad der nicht-funktionalen Anforderungen an das Instrumentenkabel

Nr.	Priorität	Anforderung	Erfüllt
K-2	2	Die Anschlusskontakte müssen so montiert sein, dass sie Spiel und Bedienung der E-Gitarre nicht beeinflussen.	Teilweise
K-3	2	Es muss erkennbar sein, welcher Signalkontakt welches Tonsignal überträgt.	Ja

9.3 Vorverarbeitung

Die Beurteilung der funktionalen Anforderungen an die Vorverarbeitung erfolgt in Tabelle 9.5. Die Anforderungen V-1 bis V-6 und V-8 sind vollständig erfüllt. Die Anforderung V-7 ist teilweise erfüllt.

V-7: Das Audiointerface kann nicht alle seine acht Eingänge direkt auf die Ausgänge weiterleiten. Es ist aber möglich in der DAW-Software dies einzustellen. Die Signale haben dann aber keine 0 ms Latenz sondern werden digitalisiert und dann wieder in Analogsignale gewandelt.

Tabelle 9.5: Erfüllungsgrad der funktionalen Anforderungen an die Vorverarbeitung

Nr.	Priorität	Anforderung	Erfüllt
V-1	2	Die Vorverstärkung muss die elektrischen Signale auf Line-Pegel verstärken.	Ja
V-2	2	Die Vorverarbeitung muss die elektrischen Signale mit mindestens 2350 Hz abtasten.	Ja
V-3	2	Die Vorverarbeitung muss die analogen Signale digitalisieren.	Ja
V-4	2	Die Vorverarbeitung muss die digitalen Signale an einen Computer übertragen.	Ja
V-5	0	Nutzende können die elektrischen Signale nach der Vorverarbeitung manuell auspegeln.	Ja
V-6	0	Die Vorverarbeitung muss ein visuelles Feedback zum Auspegeln bieten.	Ja
V-7	1	Die Vorverarbeitung bietet die Möglichkeit, die elektrischen Signale einzeln parallel abzugreifen.	Teilweise
V-8	0	Die Vorverarbeitung bietet die Möglichkeit, die elektrischen Signale zusammengefasst parallel abzugreifen.	Ja

9.4 MIDI-Konvertierung

Die Beurteilung der funktionalen Anforderungen an die MIDI-Konvertierung erfolgt in Tabelle 9.6 und die Beurteilung der nicht-funktionalen Anforderungen erfolgt in Tabelle 9.7. Die Anforderungen S-3, S-6, S-7, S-12 und S-13 sind vollständig erfüllt. Die Anforderungen S-1 und S-2 ist teilweise erfüllt. Die Anforderungen S-4, S-5, S-8 bis S-11 und S-14 sind ausstehend.

S-1: Der Audio-Editor „Wavosaur“ beinhaltet kein Multitracking, das heißt er kann nur eine Signalquelle zur Zeit aufnehmen. Das Audiointerface bietet die Option, die Eingänge 1 und 2 als eine Stereoquelle mit zwei getrennten Tonkanälen aufzunehmen, sodass die E- und A-Saite gleichzeitig in „Wavosaur“ aufgezeichnet werden können. Mehr Signalquellen

können aber nicht gleichzeitig aufgenommen werden. Auf dem Markt gibt es diverse alternative kostenpflichtige Anwendungen, die Multitracking beinhalten.

S-2: Wie in Kapitel 8.2.9 erwähnt, kann die Funktion *yin* monophone Melodien erkennen und die Töne den Saiten zuordnen. Sie hat aber vor allem in Intervallen und Akkorden Probleme, die höheren Töne zu erkennen und kann deswegen polyphone Melodien nur teilweise korrekt erfassen.

Tabelle 9.6: Erfüllungsgrad der funktionalen Anforderungen an die MIDI-Konvertierung

Nr.	Priorität	Anforderung	Erfüllt
S-1	1	Die Software muss die Möglichkeiten zur Speicherung der Rohdaten als WAV-Datei bieten.	Teilweise
S-2	2	Die Software muss anhand der elektrischen Signale erkennen können, welche Töne wie laut und wie lange auf welcher Saite gespielt wurden.	Teilweise
S-3	2	Nutzende können den erwarteten Frequenzbereich, die Analysefenstergröße und die Analyseschrittweite ändern.	Ja
S-4	1	Über die Bedienoberfläche können Nutzende einstellen, ob eine Tabulatur erzeugt und/oder eine MIDI-Konvertierung durchgeführt wird.	Ausstehend
S-5	1	Über die Bedienoberfläche können Nutzende einstellen, ob MIDI-Daten gespeichert und/oder live ausgegeben werden sollen.	Ausstehend
S-6	2	Die Software kann die analysierten Signale in das MIDI-Format konvertieren.	Ja
S-7	2	Die Software muss die MIDI-Daten im Format .mid abspeichern.	Ja
S-8	0	Die Software kann die MIDI-Daten nach Erzeugung ausgeben.	Ausstehend
S-9	0	Die Software kann eine Tabulatur erzeugen.	Ausstehend
S-10	0	Die Software kann die erzeugte Tabulatur im Format .xml abspeichern.	Ausstehend

Tabelle 9.7: Erfüllungsgrad der nicht-funktionalen Anforderungen an die MIDI-Konvertierung

Nr.	Priorität	Anforderung	Erfüllt
S-11	1	Die Software für die MIDI-Konvertierung und die Tabulatur-Erzeugung besteht aus zwei separaten Software-Modulen.	Ausstehend
S-12	2	Die Software muss auf einem Windows-Betriebssystem ab Version 10 lauffähig sein.	Ja
S-13	2	Bei der Verwendung des MIDI-Formats müssen die technischen Spezifikationen der MMA für MIDI 1.0 beachtet werden.	Ja
S-14	2	Bei der Verwendung des MusicXML-Formats müssen die technischen Spezifikationen für MusicXML Version 4.0 beachtet werden.	Ausstehend

9.5 Allgemein

Die Beurteilung der nicht-funktionalen Anforderungen an das gesamte System erfolgt in Tabelle 9.8. Die Anforderungen A-1 ist vollständig erfüllt. Die Anforderung A-2 ist nicht erfüllt.

A-2: Aktuell werden manuell WAV-Dateien aufgenommen, gespeichert und in GNU Octave eingelesen. Dementsprechend existiert keine Live-Verarbeitung. Wie in Kapitel 8.4 erläutert, wäre schon die Zeit um die minimal benötigten Samples live aufzunehmen größer als die Latenzgrenze von 16 ms.

Tabelle 9.8: Nicht-funktionale Anforderungen an das gesamte System

Nr.	Priorität	Anforderung	Erfüllt
A-1	2	Für jedes eingebaute Bauteil müssen sein Zweck, seine Bedienung und mögliche einstellbare Parameter dokumentiert sein.	Ja
A-2	1	Die Verarbeitungskette für die Echtzeit-MIDI-Ausgabe darf eine Latenz von 16 ms nicht überschreiten.	Nein

10 Fazit und Ausblick

In dieser Arbeit wurde ein hexaphonischer Tonabnehmer für E-Gitarren entwickelt und die mit ihm aufgenommenen Tonspuren mithilfe eines Algorithmus für Tonhöhenerkennung in eine MIDI-Datei konvertiert.

10.1 Hexaphonischer Tonabnehmer

Der Bau des hexaphonischen Tonabnehmers hat sich deutlich schwieriger gestaltet als angenommen. Die größten Probleme und die daraus abgeleiteten Verbesserungen bei der Weiterentwicklung des betreuenden Erstprüfers werden im Folgenden zusammenfasst.

- Die Befestigung der Tonabnehmerspule in einer Wellenkupplung ist instabil, die Spule kippt leicht beim Wickeln und der Kupferlackdraht reißt
 - => Verbesserung: Aufbau einer Spulenwickelmaschine, in der Einzelgehäuse für Spulen sicher fixiert und umwickelt werden können
- Spulenmagnete mit gleicher Polausrichtung stoßen sich innerhalb des Tonabnehmergehäuses gegenseitig ab und drücken sich aus ihren Steckplätzen
 - => Verbesserung: Abwechselnde Polausrichtung und abwechselnder Wicklungssinn, um die Richtung der Ausgangsspannung beizubehalten
- Tonabnehmerspulen mit Wachs beträufeln führt zu einer nur oberflächlichen Wachabdeckung und keiner Durchdringung der Wicklungslagen
 - => Verbesserung: Spulen einzeln auf einem Löffel in ein Wachbad tauchen, bis ihre Wicklungslagen vollständig durchdrungen sind

- Montage des Single Coil-Gehäuses auf dem Humbucker-Montageblech ist schwierig, da das Gehäuse zu breit ist und das Montageblech die elektrischen Anschlusskontakte kurzschließen kann
- => Verbesserung: Entwicklung einer Platine mit passenden Montagelöchern und elektrischen Kontaktanschlüssen

Da der Tonabnehmer aktuell nur zwei funktionierende Spulen besitzt, wird er in Zukunft durch die Weiterentwicklung des betreuenden Erstprüfers ersetzt, siehe Abbildung 10.1. Seine genutzten 3D-Druck-Dateien, Platinen-Entwürfe und sein Programmcode zum Spulenwickeln sind in seinem GitHub-Repository zu finden.¹ Aktuell treten weiterhin Probleme bei der Entfernung der Lackbeschichtung des Kupferlackdrahts auf. Dafür wird in Zukunft ein Verfahren aus der Industrie ausprobiert, bei dem der Lack mit einer Mischung aus Brennspiritus und Wasser abgetragen wird, ohne den innen liegenden Kupferdraht zu beschädigen.

10.2 MIDI-Konvertierung

Für die MIDI-Konvertierung muss tiefer gehende Nachbearbeitung in der Software integriert werden, um doppelt aufgenommene Töne und Artefakte besser entfernen zu können. Außerdem können die Schwellwerte für die maximale Aperiodizität und die minimale Signalleistung nicht bei statischen Werten bleiben, sondern müssen sich dynamisch an den Inhalt der Tonspuren anpassen können. Losgelöst vom YIN-Algorithmus sollten auch aktuelle Ansätze auf dem Gebiet der neuronalen Netze und künstlicher Intelligenz in Betracht gezogen werden. In Zukunft kann die MIDI-Konvertierung auf einen Mikrocontroller oder einen FPGA ausgelagert werden, um eine geringere Latenz zu erreichen.

¹<https://github.com/MarcOnTheMoon/>

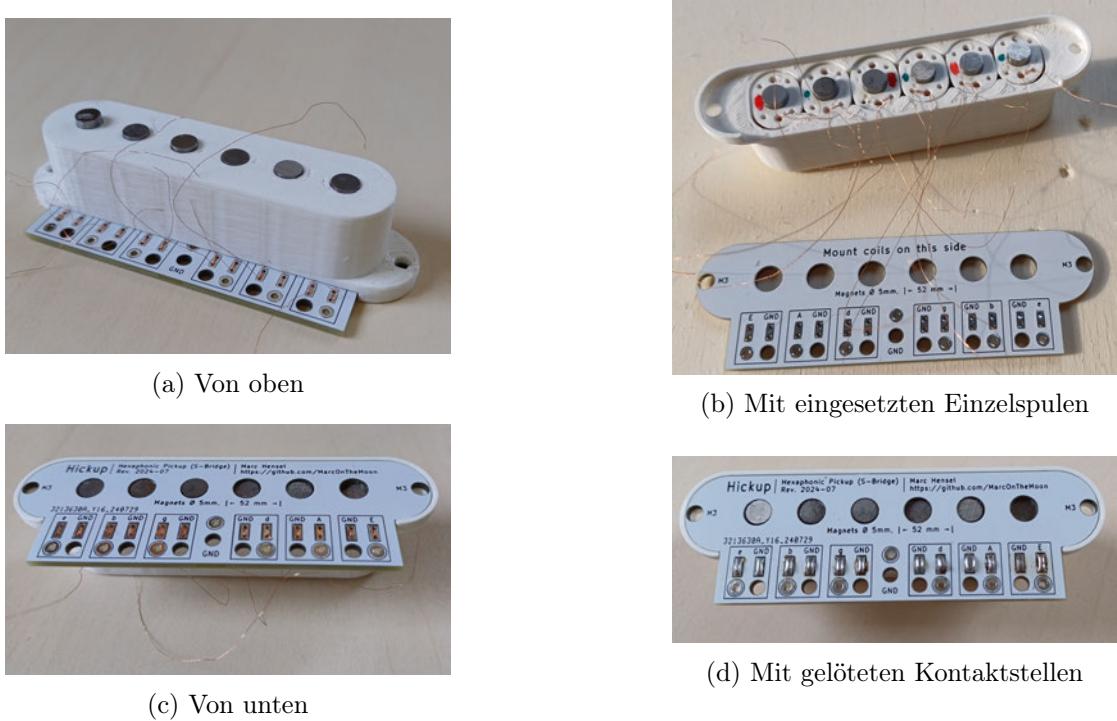


Abbildung 10.1: Zusammengebauter Tonabnehmer des betreuenden Erstprüfers

Literaturverzeichnis

- [1] BABACAN, Onur ; DRUGMAN, Thomas ; D’ALESSANDRO, Nicolas ; HENRICH, Nathalie ; DUTOIT, Thierry: A comparative study of pitch extraction algorithms on a large variety of singing sounds. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, S. 7815–7819
- [2] BITTNER, Rachel M. ; BOSCH, Juan J. ; RUBINSTEIN, David ; MESEGUE-BROCAL, Gabriel ; EWERT, Sebastian: A Lightweight Instrument-Agnostic Model for Polyphonic Note Transcription and Multipitch Estimation. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022, S. 781–785
- [3] BOGERT, Bruce P.: The Quefrency Alanysis of Time Series for Echoes: Cepstrum, Pseudo-Autocovariance, Cross-Cepstrum and Saphe Cracking. In: *Proceedings of the Symposium on Time Series Analysis*, 1963, S. 209–243
- [4] COCCOLUTO, Davide ; CESARINI, Valerio ; COSTANTINI, Giovanni: OneBitPitch (OBP): Ultra-High-Speed Pitch Detection Algorithm Based on One-Bit Quantization and Modified Autocorrelation. In: *Applied Sciences* 13 (2023), Nr. 14
- [5] CYCFI RESEARCH: *Nu Series Modular Active Pickups.* – URL <https://www.cycfi.com/projects/nu-series/>. – Zugriffsdatum: 2024-07-15
- [6] DAY, Paul ; WALDENMAIER, André: *E-Gitarren: Alles über Konstruktion und Historie*. GC Carstensen Verlag, 2007. – ISBN 3-910098-20-7
- [7] DE CHEVEIGNÉ, Alain: *YIN pitch estimator*. Juli 2003. – URL <http://audition.ens.fr/adc/sw/yin.zip>. – Zugriffsdatum: 2024-06-19
- [8] DE CHEVEIGNÉ, Alain ; KAWAHARA, Hideki: YIN, a fundamental frequency estimator for speech and music. In: *The Journal of the Acoustical Society of America* 111 (2002), Nr. 4, S. 1917–1930

- [9] DIN E.V.: *DIN EN IEC 61938:2018-10, Multimedia Systeme –Leitfaden für empfohlene Charakteristika analoger Schnittstellen zur Erreichung von Kompatibilität.* Beuth-Verlag, Oktober 2018
- [10] DRUGMAN, Thomas ; HUYBRECHTS, Goeric ; KLIMKOV, Viacheslav ; MOINET, Alexis: Traditional Machine Learning for Pitch Detection. In: *IEEE Signal Processing Letters* 25 (2018)
- [11] EICHLER, Jürgen ; MODLER, Andreas: *Physik für das Ingenieurstudium.* Springer Vieweg, April 2023. – ISBN 978-3-658-38834-8
- [12] ELLIS, Alexander J.: On the Musical Scales of Various Nations. In: *The Journal of the Society of Arts* 33 (1885), Nr. 1688, S. 485–527
- [13] FISHMAN: *Triplay Play Wireless.* – URL <https://www.fishman.com/portfolio/tripleplay-wireless-midi-guitar-controller/>. – Zugriffsdatum: 2024-07-15
- [14] FLOTZINGER, Rudolf: Lautentabulatur. In: *Oesterreichisches Musiklexikon online* (2004)
- [15] FOGG, Rod: *Das E-Gitarren Handbuch.* Voggenreiter Verlag OHG, 2011. – ISBN 9783802408359
- [16] GERHARD, David: Pitch Extraction and Fundamental Frequency: History and Current Techniques / Department of Computer Science, University of Regina, Regina, Canada. November 2003. – Forschungsbericht
- [17] GOOD, Michael: MusicXML for notation and analysis. In: *The virtual score: representation, retrieval, restoration* 12 (2001), Nr. 113-124, S. 160
- [18] GRAPH TECH GUITAR LABS: *ghost Pickups.* – URL <https://graphtech.com/pages/ghost-pickup-systems>. – Zugriffsdatum: 2024-07-15
- [19] HARTEN, Ulrich: *Physik: Eine Einführung für Naturwissenschaftler und Ingenieure.* Springer Vieweg, September 2021. – ISBN 978-3-662-61698-7
- [20] HELMHOLTZ, Hermann: *Die Lehre von den Tonempfindungen als physiologische Grundlage für die Theorie der Musik.* Vieweg, 1863
- [21] HUFSCHEID, Markus: *Grundlagen der Elektrotechnik: Einführung für Studierende der Ingenieur- und Naturwissenschaften.* Springer Vieweg, Januar 2021. – ISBN 978-3-658-30386-0

- [22] JAHNEL, Franz: *Die Gitarre und ihr Bau: das umfassende Praxishandbuch für alle Bauformen*. Ed. Bochinsky in der PPVMedien GmbH, 2008. – ISBN 978-3-923639-09-0
- [23] JAM ORIGIN: *MIDI Guitar 2.* – URL <https://www.jamorigin.com/>. – Zugriffsdatum: 2024-07-15
- [24] KEDEM, Benjamin: Spectral analysis and discrimination by zero-crossings. In: *Proceedings of the IEEE* 74 (1986), Nr. 11, S. 1477–1493
- [25] LEMME, Helmuth: *Elektrogitarren -Technik & Sound*. Elektor-Verlag, 2006. – ISBN 978-3-89576-111-9
- [26] LEMME, Helmuth: *Pickups, Potis & Co.* PPVMEDIEN, 2020. – ISBN 978-3-95512-212-6
- [27] LESTER, Michael ; BOLEY, Jon: The Effects of Latency on Live Sound Monitoring. In: *Audio engineering society convention 123*, Oktober 2007
- [28] LOCKWOOD, Dave: Blue Chip Axon AX100. In: *Sound On Sound* (1998). – URL <https://www.soundonsound.com/reviews/blue-chip-axon-ax100>
- [29] LOVER, Seth E.: *Magnetic Pickup For Stringed Musical Instrument*. U.S. Patent 2,896,491, Juli 1959
- [30] MAUCH, Matthias ; CANNAM, Chris ; BITTNER, Rachel ; FAZEKAS, George ; SALAMON, Justin ; DAI, Jiajie ; BELLO, Juan ; DIXON, Simon: Computer-aided Melody Note Transcription Using the Tony Software: Accuracy and Efficiency. In: *Proceedings of the First International Conference on Technologies for Music Notation and Representation*, URL <https://code.soundsoftware.ac.uk/projects/topy>, May 2015
- [31] MAUCH, Matthias ; DIXON, Simon: PYIN: A fundamental frequency estimator using probabilistic threshold distributions. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, S. 659–663
- [32] MiGiC: *MiGiC*. – URL <https://migic.com/>. – Zugriffsdatum: 2024-07-15
- [33] MUSIC TRIBE GLOBAL BRANDS LTD.: *U-PHORIA UMC-Series Driver-ReadMe*, 2023
- [34] PAUL RUBENSTEIN/UBERTAR: *Ubertar Hexaphonic Pickups*. – URL <https://ubertar.com/hexaphonic/products.html>. – Zugriffsdatum: 2024-07-15

- [35] PEETERS, Geoffroy: Music Pitch Representation by Periodicity Measures Based on Combined Temporal and Spectral Representations. In: *IEEE International Conference on Acoustics Speech and Signal Processing Proceedings* Bd. 5, 2006, S. V–V
- [36] PISZCZALSKI, Martin ; GALLER, Bernard A.: Predicting musical pitch from component frequency ratios. In: *The Journal of the Acoustical Society of America* 66 (1979), Nr. 3, S. 710–720
- [37] RAO, Prajwal S. ; KHOUSHIKH, S ; RAVISHANKAR, Sriram ; ANANTHKRISHNAN, R A. ; BALACHANDRA, K: A Comparative Study of Various Pitch Detection Algorithms. In: *2020 5th International Conference on Computing, Communication and Security (ICCCS)*, 2020, S. 1–6
- [38] RHODE & SCHWARZ: *Audio Analyzer R&S UPV*, Juli 2006
- [39] ROADS, Curtis: *The Computer Music Tutorial*. 1. Edition. MIT Press, 1996. – ISBN 0-262-68082-3
- [40] ROBJOHNS, Hugh: Understanding Imepdiance. In: *Sound On Sound* (2003). – URL <https://www.soundonsound.com/techniques/understanding-impedance>
- [41] ROLAND CORPORATION: *GK-3*. – URL <https://www.roland.com/de/products/gk-3/>. – Zugriffsdatum: 2024-07-15
- [42] SALAMON, Justin ; GOMEZ, Emilia: Melody Extraction From Polyphonic Music Signals Using Pitch Contour Characteristics. In: *IEEE Transactions on Audio, Speech, and Language Processing* 20 (2012), Nr. 6, S. 1759–1770
- [43] SCHaedla, Ulf: Der elektromagnetische Tonabnehmer als Wandler. In: *Guitar Letters I* (2006)
- [44] SHANNON, C.E.: Communication in the Presence of Noise. In: *Proceedings of the IRE* 37 (1949), Nr. 1, S. 10–21
- [45] SINAMBARI, Gh. R. ; SENTPALI, Stefan: *Ingenieurakustik*. Springer Vieweg, 2020. – ISBN 978-3-658-27288-3
- [46] THE MIDI MANUFACTURERS ASSOCIATION: *General MIDI System Level 1*, 1991
- [47] THE MIDI MANUFACTURERS ASSOCIATION: *MIDI 1.0 Detailed Specification*. Version 4.2.1, Februar 1996

- [48] THE MIDI MANUFACTURERS ASSOCIATION: *Standard MIDI files 1.0*, Februar 1996
- [49] WEINZIERL, Stefan (Hrsg.): *Handbuch der Audiotechnik*. Springer Berlin, Heidelberg, Januar 2009. – ISBN 978-3-540-34301-1
- [50] WHITE, Paul: Roland GI-10. In: *Sound On Sound* (1995). – URL <https://www.soundonsound.com/reviews/roland-gi-10>
- [51] WHITE, Paul: Yamaha G50. In: *Sound On Sound* (1996). – URL <https://www.soundonsound.com/reviews/yamaha-g50>
- [52] WHITE, Paul: Jam Origin MIDI Guitar 2. In: *Sound On Sound* (2017). – URL <https://www.soundonsound.com/reviews/jam-origin-midi-guitar-2>
- [53] WINER, Ethan: *The Audio Expert: Everything You Need To Know About Audio*. Routledge, 2017. – ISBN 978-0-415-78883-0
- [54] WOLFE, Joe: *Note names, MIDI numbers and frequencies*. – URL <https://newt.phys.unsw.edu.au/jw/notes.html>. – Zugriffsdatum: 2024-08-15
- [55] XI, Qingyang ; BITTNER, Rachel M. ; PAUWELS, Johan ; YE, Xuzhou ; BELLO, Juan P.: Guitarset: A dataset for guitar transcription. In: *Proceedings of the 19th International Society for Music Information Retrieval Conference* (2018), S. 453–460. – URL <https://guitarset.weebly.com/>
- [56] ZEM COLLEGE - INSTITUT FÜR ELEKTRONISCHE MUSIK: *Notenbezeichnungen*. – URL https://www.zem-college.de/midi/mc_taben.htm. – Zugriffsdatum: 2024-08-06
- [57] ZOLLNER, Manfred: *Physik der Elektrogitarre*. Selbstverlag Manfred Zollner, 2009

A Anhang

A.1 Frequenzen der gleichstufigen Stimmung und ihre Tonnamen und MIDI-Nummern

Tabelle A.1: MIDI-Nummern und dazugehörige Tonnamen und Frequenzen [56]

MIDI-Nummer	deutscher Name	englischer Name	Frequenz in Hz
0	C,,,	C-1	8,176
1	Cis,,,/Des,,,	C♯-1/D♭-1	8,662
2	D,,,	D-1	9,177
3	Dis,,,/Es,,,	D♯-1/E♭-1	9,723
4	E,,,	E-1	10,301
5	F,,,	F-1	10,913
6	Fis,,,/Ges,,,	F♯-1/G♭-1	11,562
7	G,,,	G-1	12,250
8	Gis,,,/As,,,	G♯-1/A♭-1	12,978
9	A,,,	A-1	13,750
10	Ais,,,/B,,,	A♯-1/B♭-1	14,568
11	H,,,	B-1	15,434
12	C,,	C0	16,352
13	Cis,,,/Des,,,	C♯0/D♭0	17,324
14	D,,	D0	18,354
15	Dis,,,/Es,,,	D♯0/E♭0	19,445
16	E,,	E0	20,601
17	F,,	F0	21,826
18	Fis,,,/Ges,,,	F♯0/G♭0	23,124
19	G,,	G0	24,499

Tabelle A.1: MIDI-Nummern und dazugehörige Tonnamen und Frequenzen [56]

MIDI-Nummer	deutscher Name	englischer Name	Frequenz in Hz
20	Gis,,/As,,	G \sharp 0/A \flat 0	25,956
21	A,,	A0	27,500
22	Ais,,/B,,	A \sharp 0/B \flat 0	29,135
23	H,,	B0	30,867
24	C,	C1	32,703
25	Cis,/Des,	C \sharp 1/D \flat 1	34,648
26	D,	D1	26,708
27	Dis,/Es,	D \sharp 1/E \flat 1	38,890
28	E,	E1	41,203
29	F,	F1	43,653
30	Fis,/Ges,	F \sharp 1/G \flat 1	46,249
31	G,	G1	48,999
32	Gis,/As,	G \sharp 1/A \flat 1	51,913
33	A,	A1	55,000
34	Ais,/B,	A \sharp 1/B \flat 1	58,270
35	H,	B1	61,735
36	C	C2	65,406
37	Cis/Des	C \sharp 2/D \flat 2	69,295
38	D	D2	73,416
39	Dis/Es	D \sharp 2/E \flat 2	77,781
40	E	E2	82,406
41	F	F2	87,307
42	Fis/Ges	F \sharp 2/G \flat 2	92,499
43	G	G2	97,998
44	Gis/As	G \sharp 2/A \flat 2	103,82
45	A	A2	110,00
46	Ais/B	A \sharp 2/B \flat 2	116,56
47	H	B2	123,47
48	c	C3	130,81
49	cis/des	C \sharp 3/D \flat 3	138,59
50	d	D3	146,83

Tabelle A.1: MIDI-Nummern und dazugehörige Tonnamen und Frequenzen [56]

MIDI-Nummer	deutscher Name	englischer Name	Frequenz in Hz
51	dis/es	D♯3/E♭3	155,56
52	e	E3	164,81
53	f	F3	174,61
54	fis/ges	F♯3/G♭3	184,99
55	g	G3	195,99
56	gis/as	G♯3/A♭3	207,65
57	a	A3	220,00
58	ais/b	A♯3/B♭3	233,08
59	h	B3	246,94
60	c'	C4	261,63
61	cis'/des'	C♯4/D♭4	277,18
62	d'	D4	293,66
63	dis'/es'	D♯4/E♭4	311,13
64	e'	E4	329,63
65	f'	F4	349,23
66	fis'/ges''	F♯4/G♭4	369,99
67	g'	G4	391,99
68	gis'/as'	G♯4/A♭4	415,31
69	a'	A4	440,00
70	ais'/b'	A♯4/B♭4	466,16
71	h'	B4	493,88
72	c''	C5	523,25
73	cis''/des''	C♯5/D♭5	554,37
74	d''	D5	587,33
75	dis''/es''	D♯5/E♭5	622,25
76	e''	E5	659,26
77	f''	F5	698,46
78	fis''/ges''	F♯5/G♭5	739,99
79	g''	G5	783,99
80	gis''/as''	G♯5/A♭5	830,61
81	a''	A5	880,00

Tabelle A.1: MIDI-Nummern und dazugehörige Tonnamen und Frequenzen [56]

MIDI-Nummer	deutscher Name	englischer Name	Frequenz in Hz
82	ais''/b''	A♯5/B♭5	932,32
83	h''	B5	987,77
84	c'''	C6	1046,5
85	cis'''/des'''	C♯6/D♭6	1108,7
86	d'''	D6	1174,7
87	dis'''/es'''	D♯6/E♭6	1244,5
88	e'''	E6	1318,5
89	f'''	F6	1396,9
90	fis'''/ges'''	F♯6/G♭6	1480,0
91	g'''	G6	1568,0
92	gis'''/as'''	G♯6/A♭6	1661,2
93	a'''	A6	1760,00
94	ais'''/b'''	A♯6/B♭6	1864,7
95	h''''	B6	1975,5
96	c''''	C7	2093,0
97	cis''''/des''''	C♯7/D♭7	2217,5
98	d''''	D7	2349,3
99	dis''''/es''''	D♯7/E♭7	2489,0
100	e''''	E7	2637,0
101	f''''	F7	2793,8
102	fis''''/ges''''	F♯7/G♭7	2960,0
103	g''''	G7	3136,0
104	gis''''/as''''	G♯7/A♭7	3322,4
105	a''''	A7	3520,0
106	ais''''/b''''	A♯7/B♭7	3729,3
107	h''''	B7	3951,1
108	c''''	C8	4186,0
109	cis''''/des''''	C♯8/D♭8	4434,9
110	d''''	D8	4698,6
111	dis''''/es''''	D♯8/E♭8	4978,0
112	e''''	E8	5274,0

Tabelle A.1: MIDI-Nummern und dazugehörige Tonnamen und Frequenzen [56]

MIDI-Nummer	deutscher Name	englischer Name	Frequenz in Hz
113	f'''''	F8	5587,7
114	fis''''/ges'''''	F♯8/G♭8	5919,9
115	g''''	G8	6271,9
116	gis''''/as'''''	G♯8/A♭8	6644,9
117	a''''	A8	7040,0
118	ais''''/b'''''	A♯8/B♭8	7458,6
119	h''''	B8	7902,1
120	c''''''	C9	8372,0
121	cis''''''/des''''''	C♯9/D♭9	8869,9
122	d''''''	D9	9397,3
123	dis''''''/es''''''	D♯9/E♭9	9956,1
124	e''''''	E9	10548,1
125	f''''''	F9	11175,3
126	fis''''''/ges''''''	F♯9/G♭9	11839,8
127	g''''''	G9	12543,9

A.2 Programmablaufdiagramme

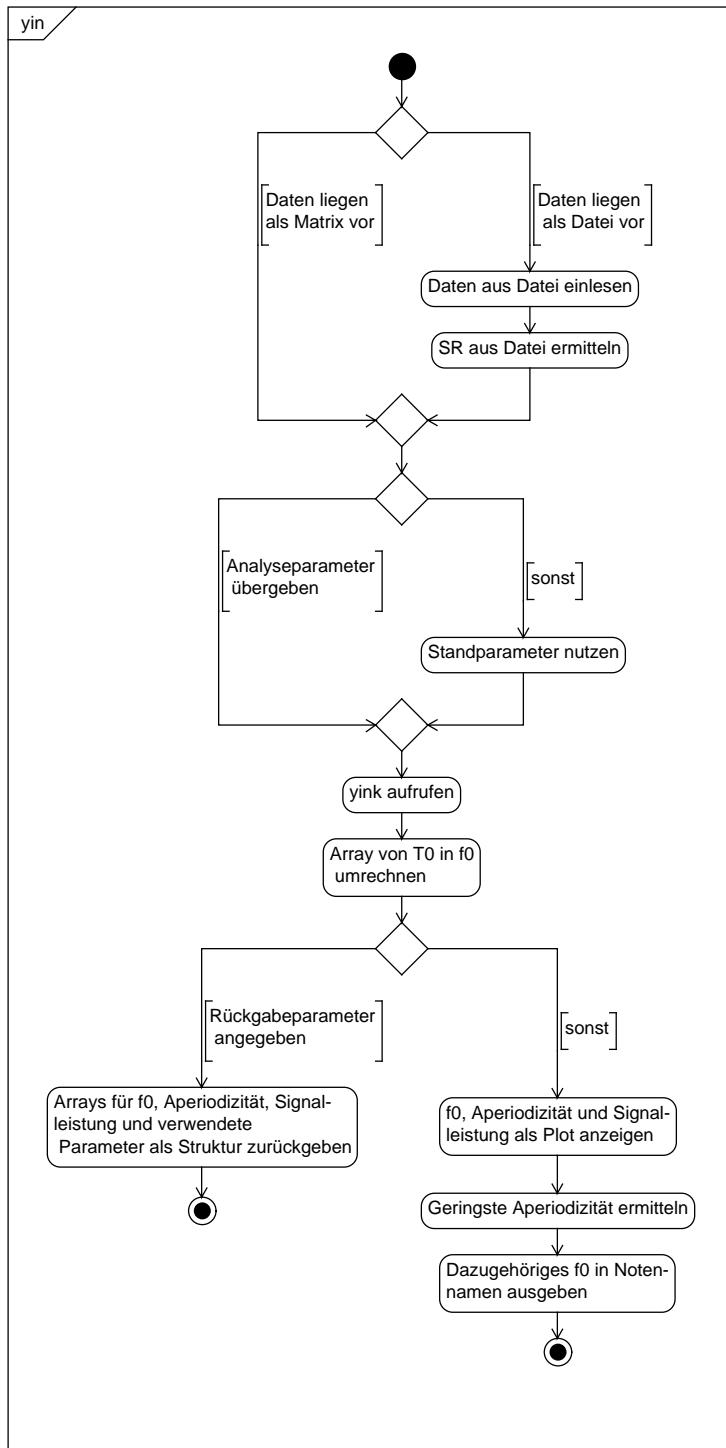


Abbildung A.1: Ablaufdiagramm der Funktion *yin*

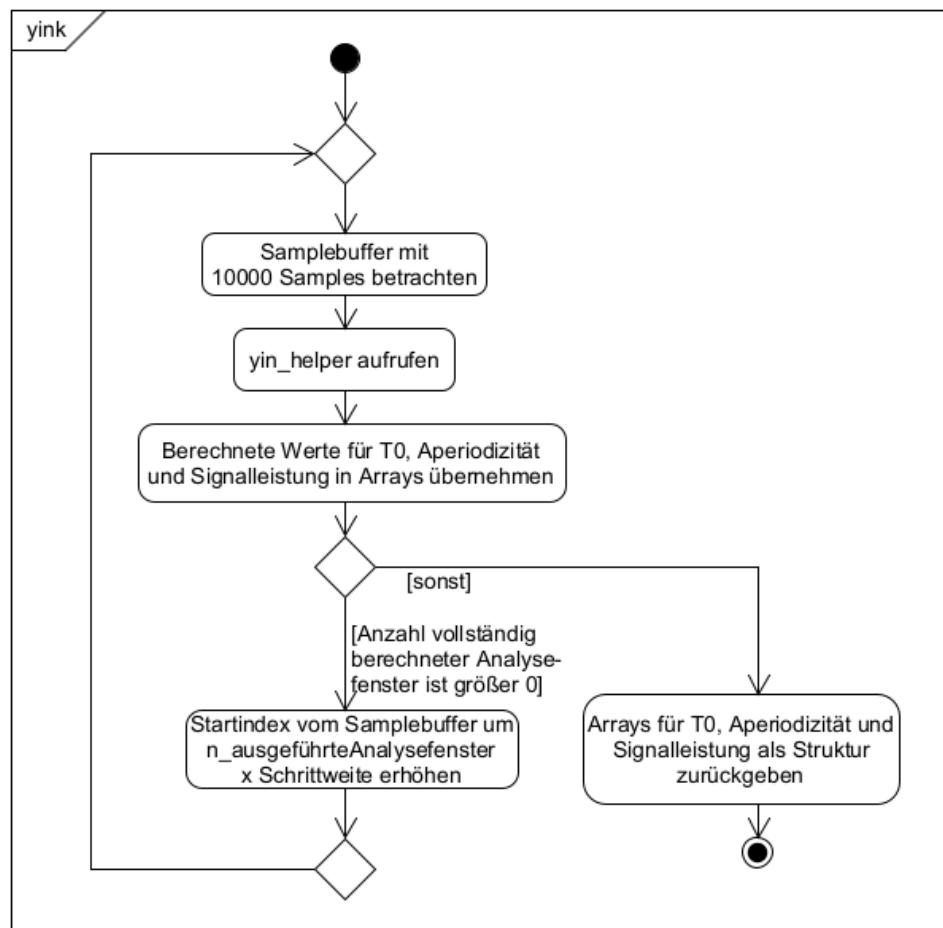


Abbildung A.2: Ablaufdiagramm der Funktion `yink`

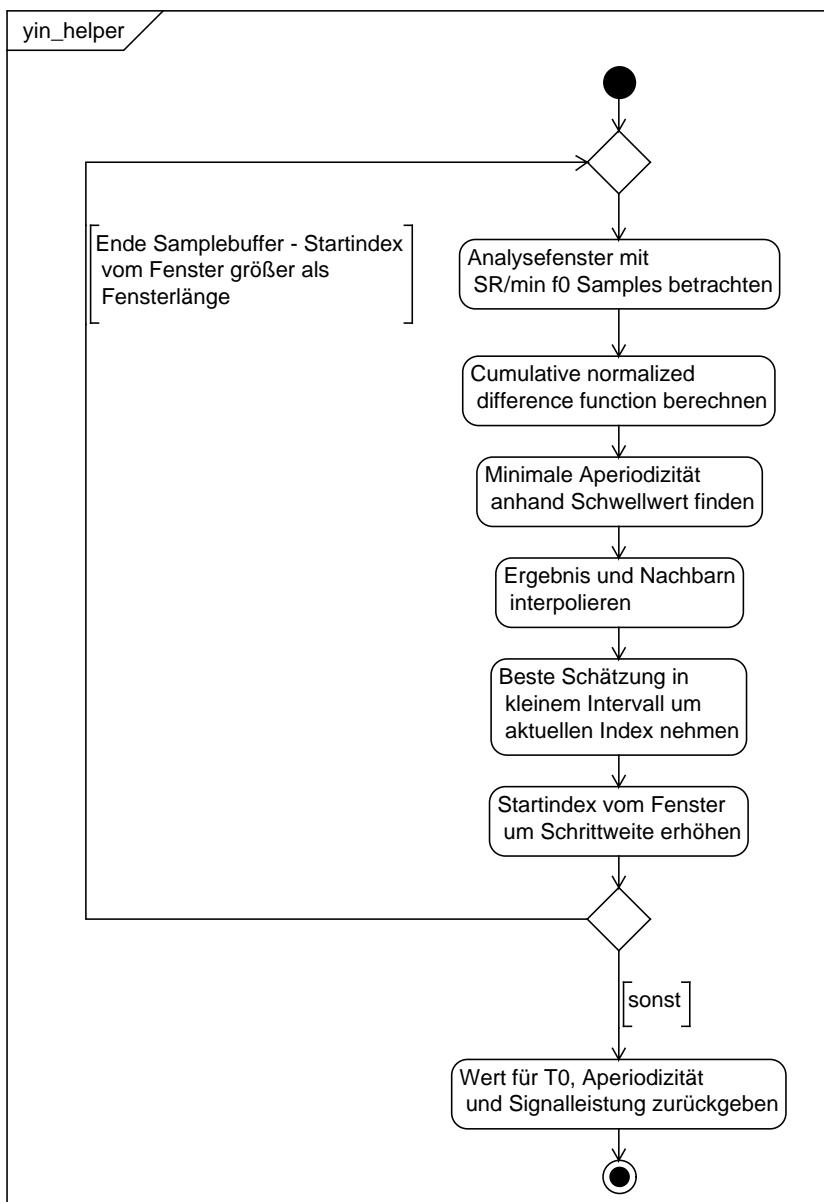


Abbildung A.3: Ablaufdiagramm der Funktion *yin_helper*

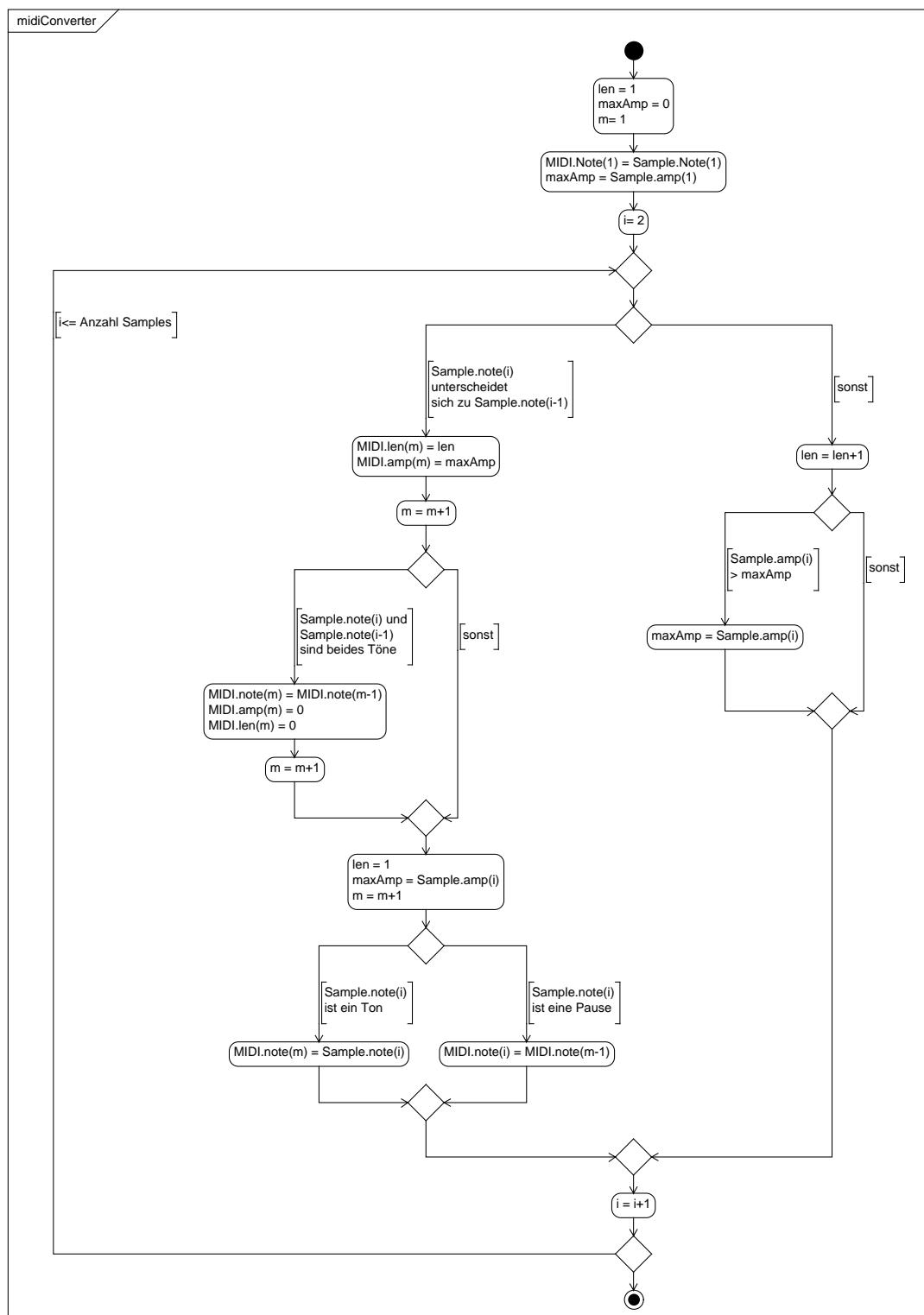
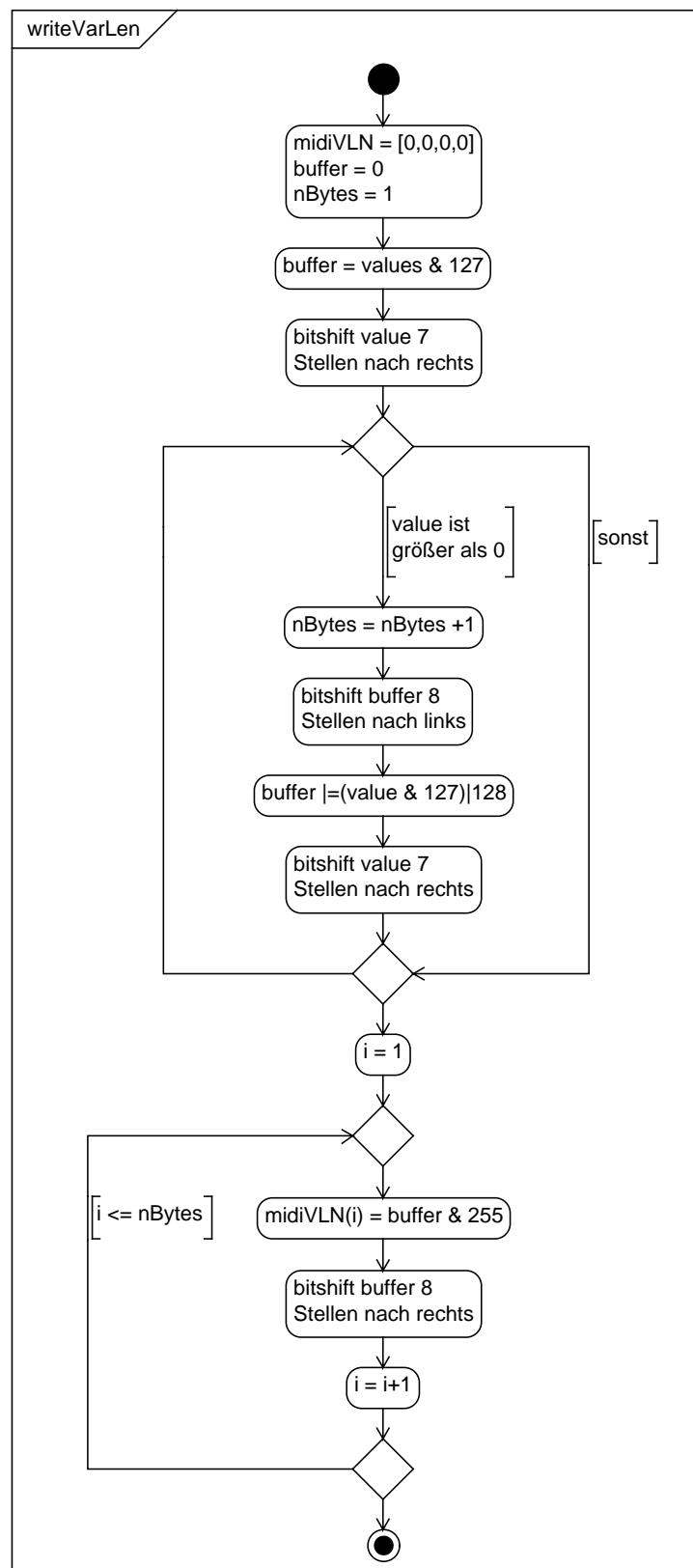


Abbildung A.4: Ablaufdiagramm der Funktion *midiConverter*

Abbildung A.5: Ablaufdiagramm der Funktion *writeVarLen*

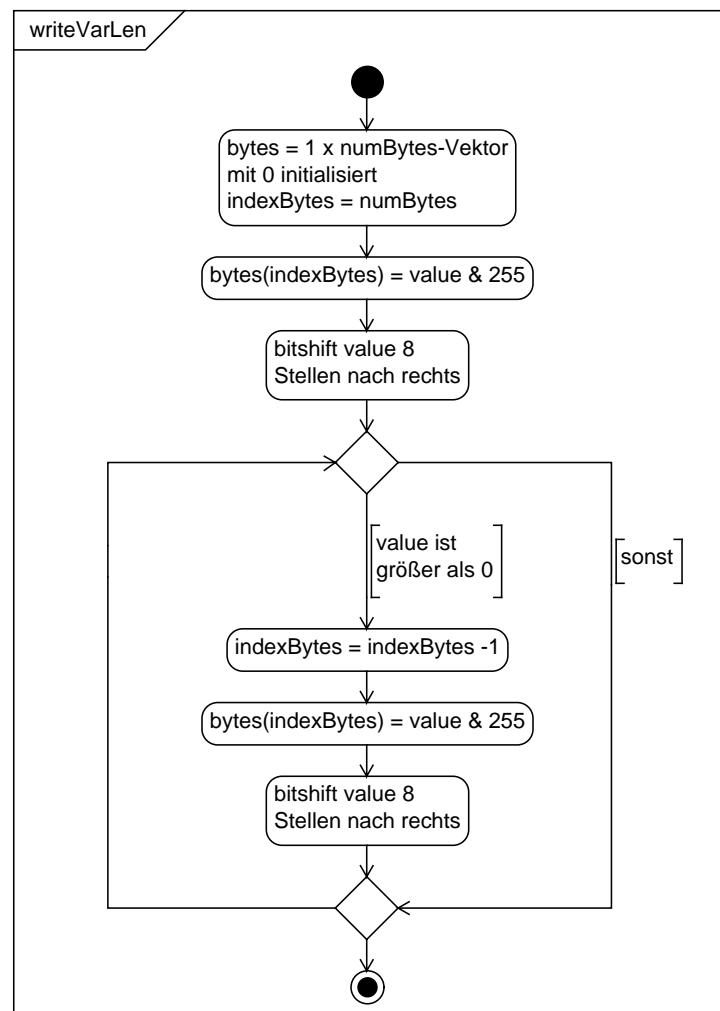


Abbildung A.6: Ablaufdiagramm der Funktion `writeBytes`

A.3 Weitere Ergebnisse der Funktion *yin*

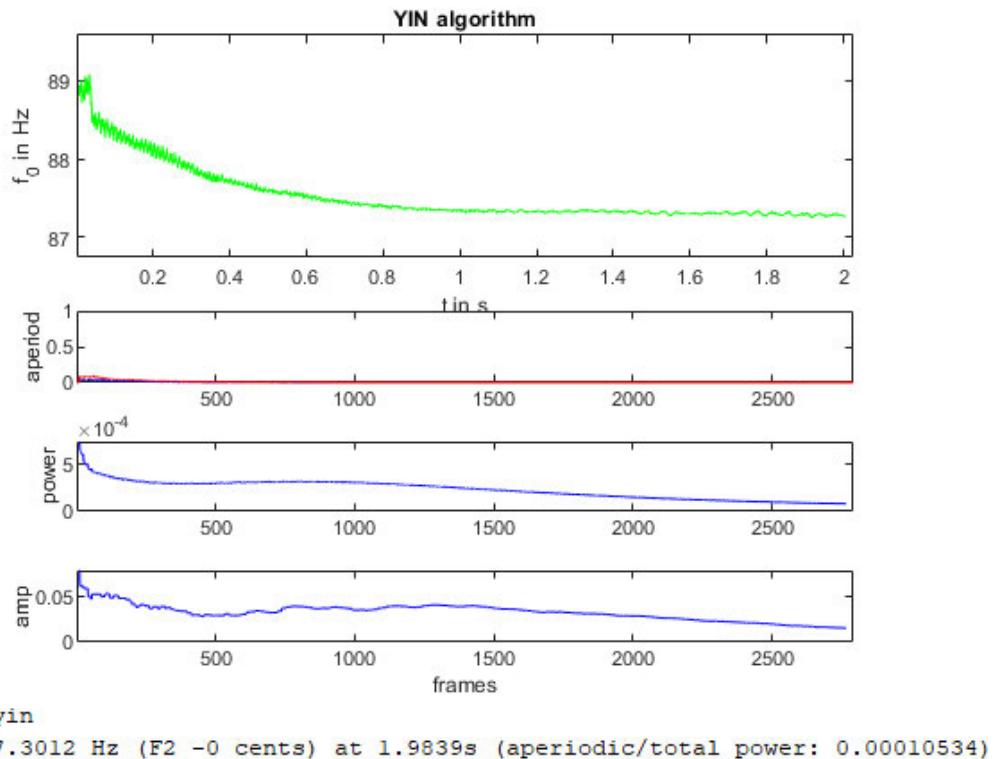


Abbildung A.7: Ausgabe der Funktion *yin* für den Ton F auf der E-Saite

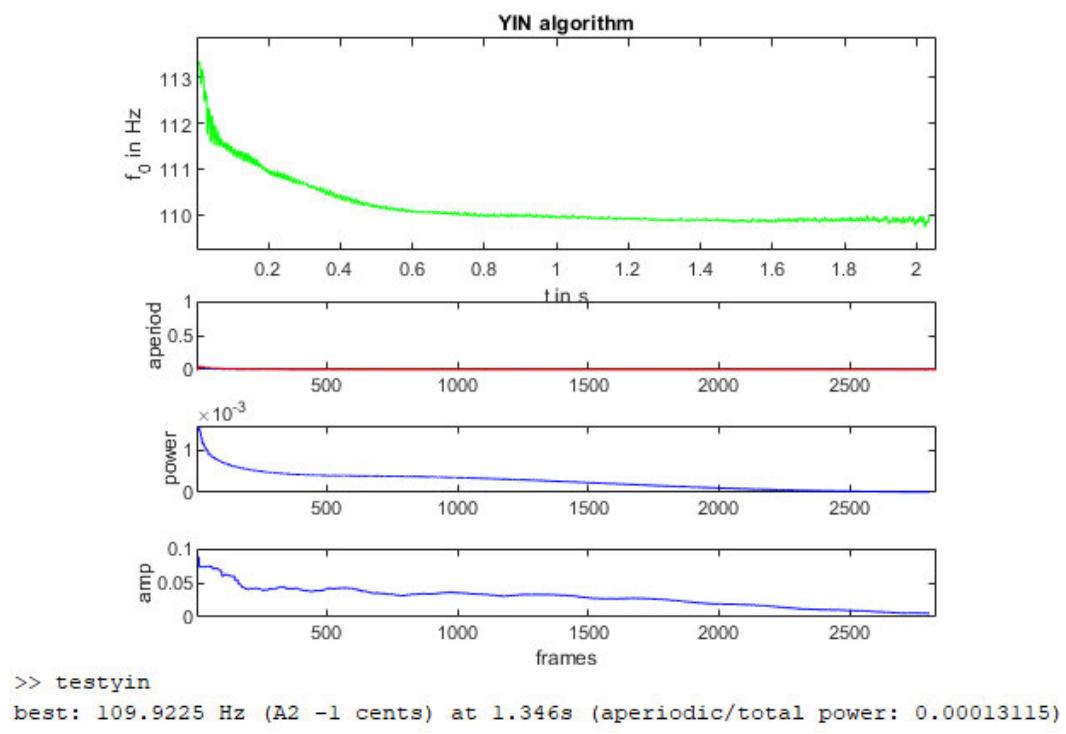


Abbildung A.8: Ausgabe der Funktion *yin* für den Ton A auf der E-Saite

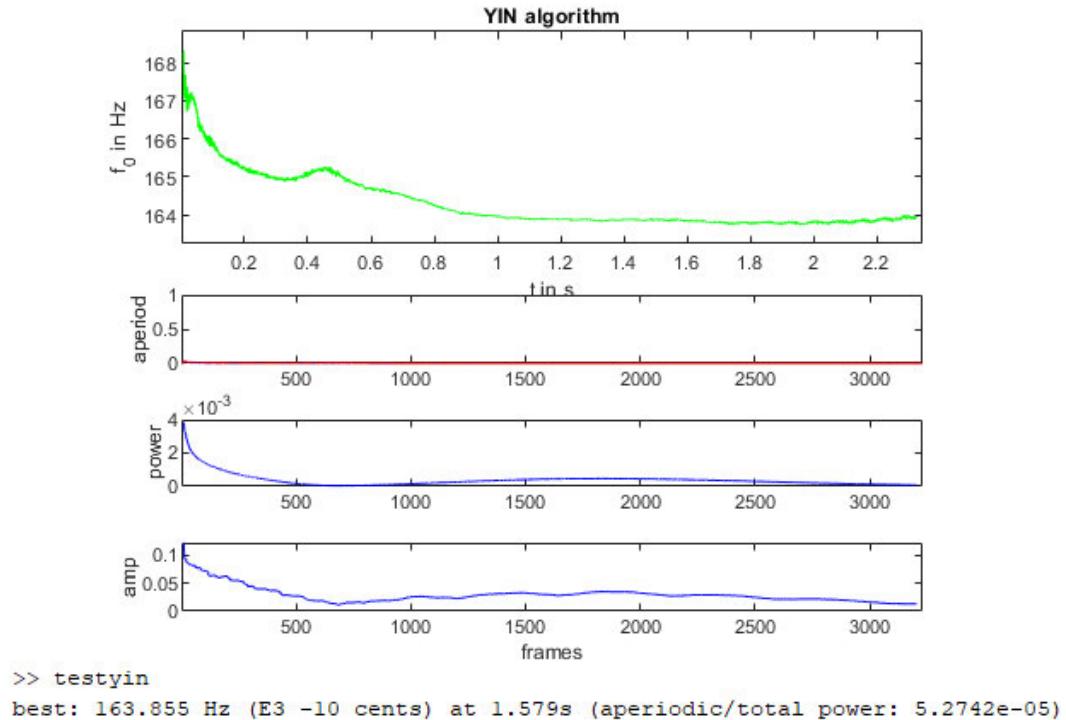


Abbildung A.9: Ausgabe der Funktion *yin* für den Ton e auf der E-Saite

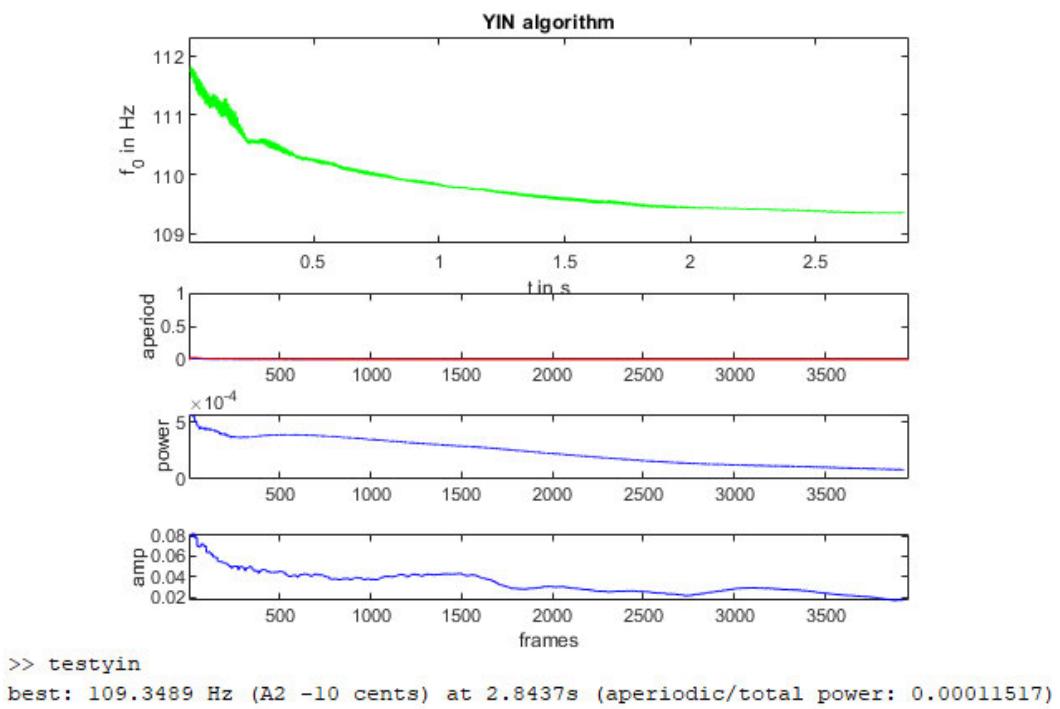


Abbildung A.10: Ausgabe der Funktion *yin* für den Ton A auf der A-Saite

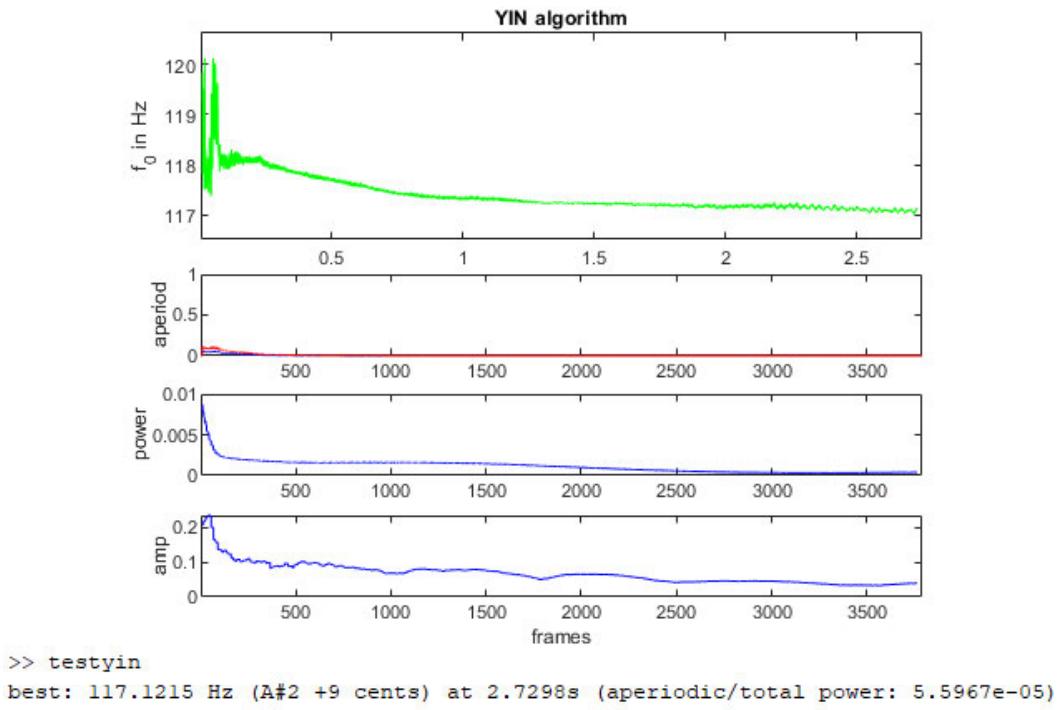


Abbildung A.11: Ausgabe der Funktion *yin* für den Ton Ais auf der A-Saite

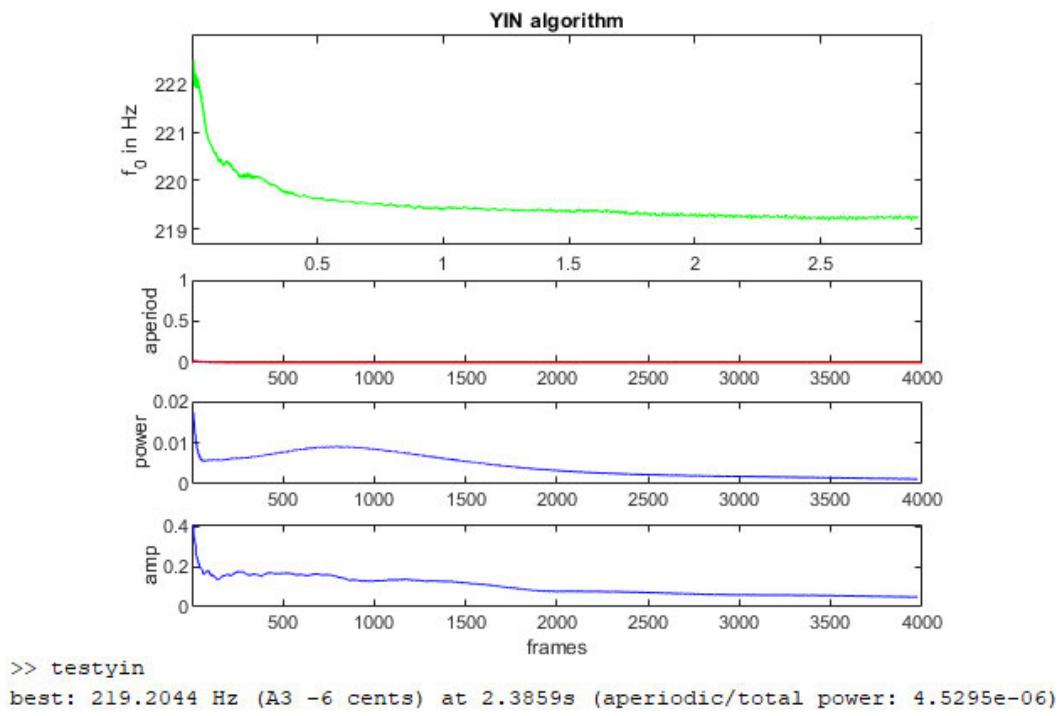


Abbildung A.12: Ausgabe der Funktion *yin* für den Ton a auf der A-Saite

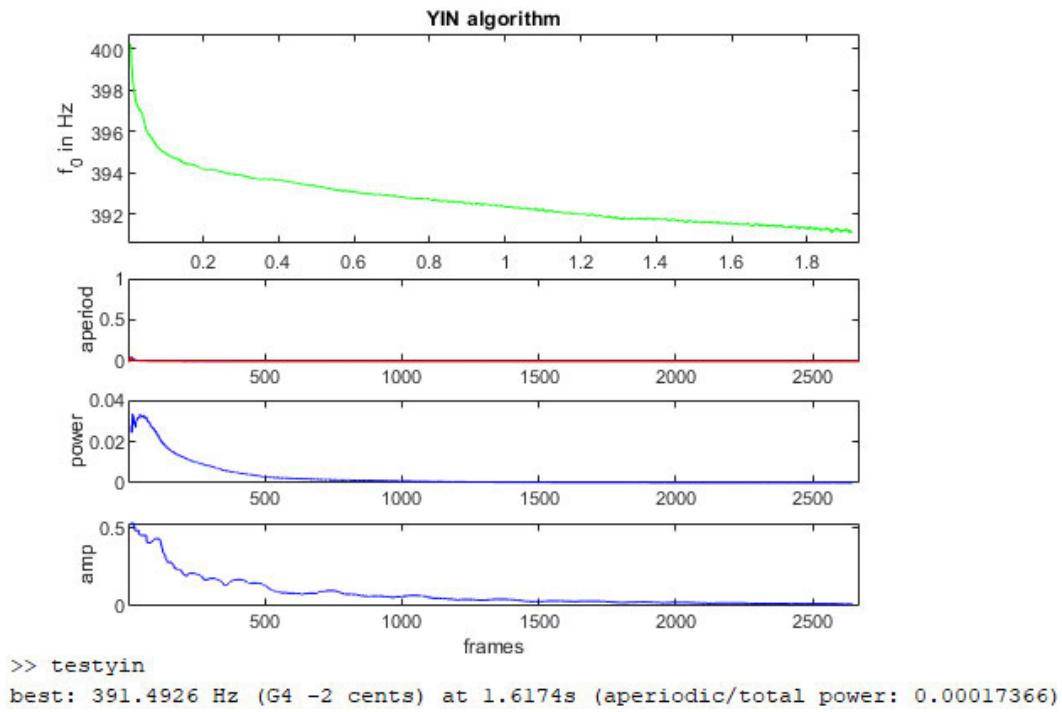


Abbildung A.13: Ausgabe der Funktion *yin* für den Ton g' auf der A-Saite

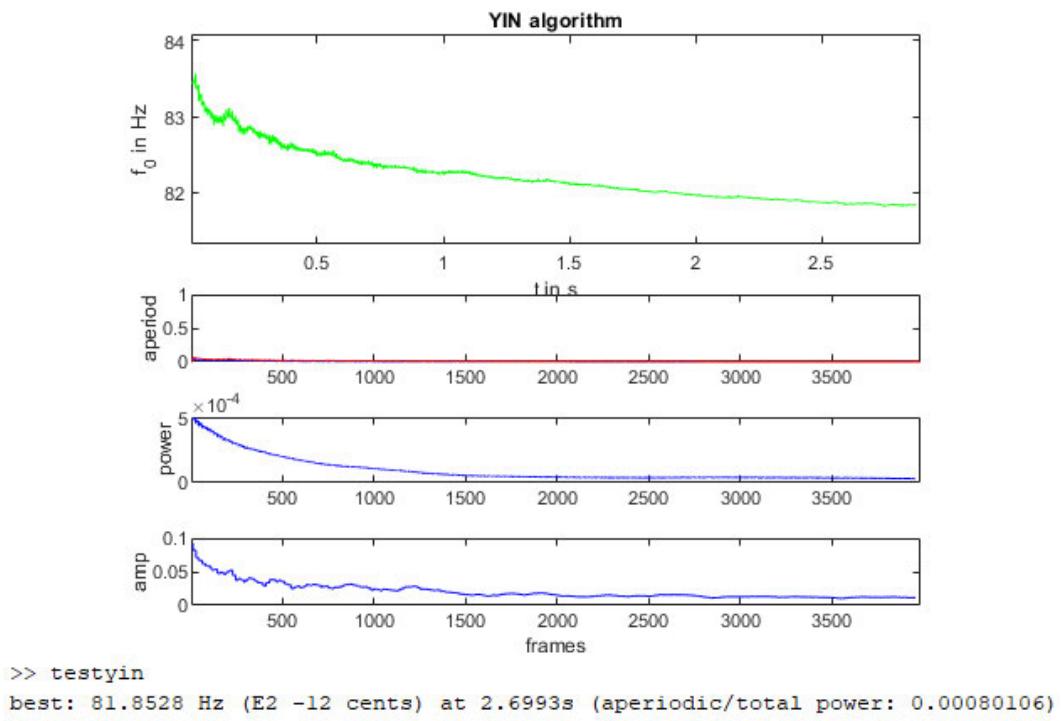


Abbildung A.14: Ausgabe der Funktion *yin* für den Ton E auf der E-Saite aufgenommen mit der A-Tonabnehmerspule

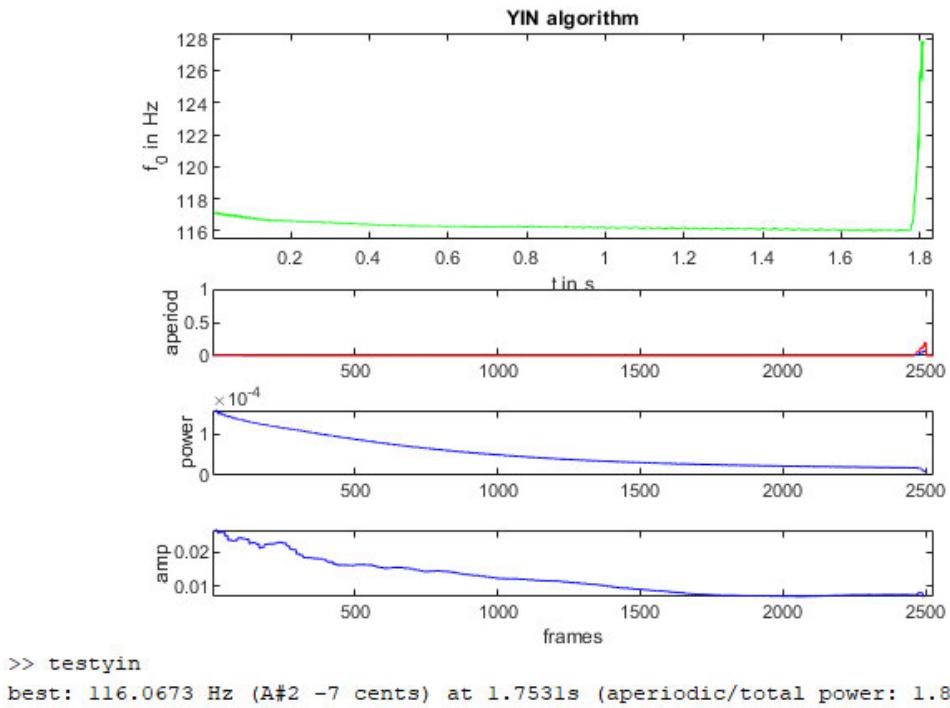
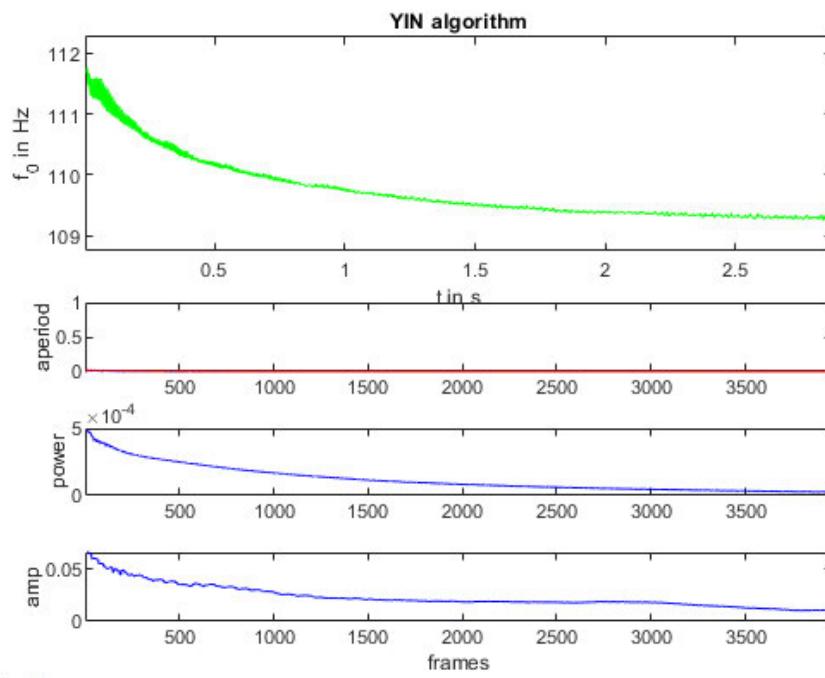
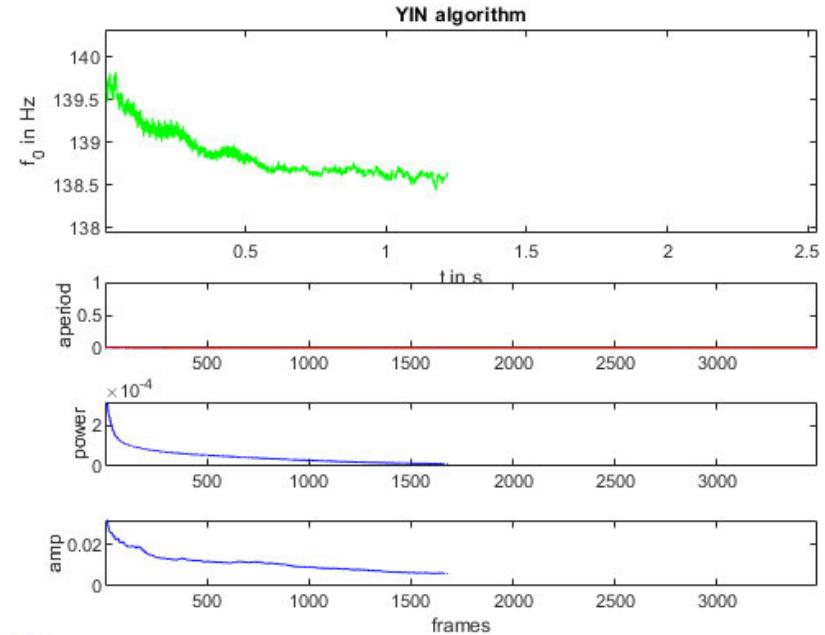


Abbildung A.15: Ausgabe der Funktion *yin* für den Ton Ais auf der E-Saite aufgenommen mit der A-Tonabnehmerspule



```
>> testyin
best: 109.4561 Hz (A2 -9 cents) at 1.645s (aperiodic/total power: 8.638e-05)
```

Abbildung A.16: Ausgabe der Funktion *yin* für den Ton A auf der A-Saite aufgenommen mit der E-Tonabnehmerspule



```
>> testyin
best: 138.6214 Hz (C#3 +0 cents) at 0.79456s (aperiodic/total power: 0.00019585)
```

Abbildung A.17: Ausgabe der Funktion *yin* für den Ton cis auf der A-Saite aufgenommen mit der E-Tonabnehmerspule

Stichwortverzeichnis

- Gitarre
 Bundstäbchen, 3
 Griffbrett, 3
 Gurt, 4
 Hals, 3
 Kopf, 3
 Korpus, 3
 Mensur, 3
 Saite, 3
 Schallloch, 3
 Steg, 3
 Stimmwirbel, 3
 Tonabnehmer, *siehe* Tonabnehmer
- Instrumentenkabel, 20
- MIDI, 24
 Datei-Header, 27
 Hardwareschnittstelle, 24
 MIDI IN, 25
 MIDI OUT, 25
 MIDI THRU, 25
 Note Off, 26
 Note On, 26
 Program Change, 27
 Track-Ende, 30
 Track-Header, 28
 Trackname, 29
 VLA, 30
- Zeitsignatur, 29
- Notationsformen, 5
 Notensystem, 5
 Tabulatur, 5
- Oberton, 8
- Oktavfehler, 40, 100
- Pegel
 Instrumenten-Pegel, 22
 Lautsprecher-Pegel, 23
 Line-Pegel, 21
 Mikrofon-Pegel, 22
- Resonanz, 15
 Güte, 15
 Überhöhung, 15
- Spieltechniken, 6
 Bending, 7
 Hammer On, 7
 Pull Off, 7
 Slide, 7
 Vibrato, 7
- Tonabnehmer, 3, 10
 Hexaphonisch, 13
 Humbucker, 12
 Piezo, 14
 Polyphon, 13

Single Coil, 11

Übertragung

Symmetrisch, 21

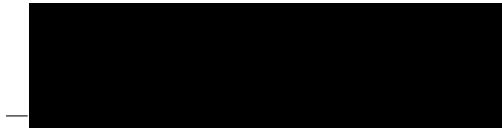
Unsymmetrisch, 21

Erklärung zur selbständigen Bearbeitung

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

Ort

Datum



Unterschrift im Original