Technische Universität Berlin
Fakultät I
Institut für Sprache und Kommunikation
Fachgebiet Audiokommunikation

Entwicklung und Evaluation eines Controllers zur Tonhöhensteuerung über Vier-Finger-Kombinationen

Masterarbeit

vorgelegt von

Gabriel Treindl

[redacted]

[redacted]

Abgabe am: 7. März 2016

Erstgutachter: Prof. Dr. Stefan Weinzierl
Zweitgutachter: Dr. Hauke Egermann
Betreuer: Henrik von Coler

Erklärung

Die selbstständige und eigenhändige Anfertigung versichere ich an Eides statt.

Gabriel Treindl

Berlin, den 3. März 2016

Danksagung

Ich möchte mich bei Prof. Dr. Stefan Weinzierl für sein Interesse an meiner Arbeit bedanken. Dr. Thomas Lerch des Staatlichen Instituts für Musikforschung danke ich recht herzlich für seine ausführliche Beratung, basierend auf seiner langjährigen Erfahrung im Bereich des Instrumentenbaus und für sein außerordentliches handwerkliches Geschick bei der mechanischen Umsetzung. Für seine intensive Unterstützung bei der Planung und Durchführung der Studie möchte ich Dr. phil. Hauke Egermann und Alexander Förstel danken. Mein besonderer Dank gilt Henrik von Coler für sein offenes Teilen von Ideen bei der Konzeption des Controllers und die intensive Beratung und Unterstützung während der gesamten Zeit der Masterarbeit. Hiermit bedanke ich mich bei allen Versuchsteilnehmern für ihr Interesse und ihre Anregungen.
Mein Dank geht außerdem an meine Familie, alle Freunde und Kommilitonen, die mich während des Masterstudiums unterstützt haben. Besonders danke ich Stephan Kasnitz und Nina Heck für ihre Unterstützung bei der Kurrektur dieser Arbeit. Ganz besondere Unterstützung habe ich von Lia Runfola erhalten, dafür danke ich ihr sehr!

Zusammenfassung

Es wurden bereits zahlreiche neuartige und innovative Controller für digitale Synthesizer entwickelt und vorgestellt (Miranda und Wanderley, 2006). Jedoch sind neben dem Keyboard nur wenige weit verbreitet (Medeiros et al., 2014). Kommerziell erhältliche Controller werden häufig dazu verwendet, um vordefinierte Klänge oder Sequenzen abzuspielen (Paine, 2013) und sind meist in den Interaktionsmöglichkeiten mit dem Klang stark limitiert. Im Rahmen der Masterarbeit wird ein Controller realisiert, der die Steuerung der Tonhöhe mit der Modulation weiterer Klangparameter vereint. Die Tonhöhe wird über vier druckempfindliche Tasten mit Aftertouch durch Fingerkombinationen ausgewählt. Die Steuerung der Oktave erfolgt mit dem Daumen der gleichen Hand. Durch Druck auf eine weich gelagerte Fläche wird der Ton ausgelöst und kontinuierlich moduliert. Durch die Gewinnung von Steuerinformationen bei der Tonhöhenauswahl und bei der Anregung, wird die expressive Interaktion mit einer monophonen Klangsynthese möglich.
Die Tonhöhensteuerung über Vier-Finger-Kombinationen bietet den Vorteil, dass die Finger auf den Tasten verweilen können. Im Vergleich zu einer Klaviatur ist es nicht notwendig bei einem Tonwechsel eine neue Position zu finden. Hingegen müssen bei manchen Fingerkombinationen mehrere Tasten simultan gedrückt werden. Bei der Evaluation wird untersucht, ob diese Art der Tonhöhensteuerung einen Vorteil gegenüber der Klaviatur bietet. Diese wird am häufigsten zur Tonhöhensteuerung bei digitalen Musikinstrumenten verwendet (Holland et al., 2013a, S. 123).
Wichtig für die Tonhöhensteuerung ist, dass diese schnell und fehlerfrei ausführbar ist, ohne dabei einzelne Töne zu benachteiligen oder bestimmte Tonarten zu bevorzugen. Ein Versuch mit Probanden soll diese Anforderungen überprüfen. Da es große Unterschiede zwischen den beiden Arten der Tonhöhensteuerung gibt, wird eine Nebenbedingung mit nur vier unterschiedlichen Tonhöhen geschaffen. Für diese ist die Spielweise beider Interfaces identisch. Der Controller wird zusätzlich in zwei unterschiedlichen Varianten untersucht. Bei der einen erfolgt das Spielen eines Tones direkt durch den Tastendruck und bei der anderen, über zusätzlichen Druck auf die obere Fläche. Über diese Nebenbedingungen soll es möglich werden, vergleichbare Ergebnisse zu erzielen.
In dem Experiment werden die unterschiedlichen Tonhöhen als visuelle Stimuli angezeigt. Die Versuchspersonen werden instruiert die Note der angezeigten Grifftabelle so schnell und fehlerfrei wie möglich zu spielen. Der Versuch wird mit der Klaviatur und den beiden Varianten des Controllers jeweils mit vier und zwölf Tonhöhen durchgeführt. Aus den aufgezeichneten Daten werden die Zeit zum Spielen der richtigen Note und die Fehlerrate ermittelt. Diese abhängigen Variablen werden auf die Einflüsse der Faktoren Interface-Variante und Anzahl der Noten untersucht. Hierfür wird jeweils eine zweifaktoriellen Varianzanalyse für wiederholtes Messen durchgeführt. Weist ein signifikanter Effekt oder eine Interaktion der Faktoren festgestellt auf Unterschiede hin, werden die Mittelwerte der Interfaces jeweils paarweise mithilfe eines t-Tests für verbundene Stichproben verglichen.
Für die Dauer um eine Note zu spielen, wird kein signifikanter Unterschied zwischen den Interface-Varianten festgestellt und es besteht keine nachweisbare Interaktion zwischen den Faktoren Interface-Variante und Anzahl der Noten. Jedoch zeigt die Varianzanalyse bei der Fehlerrate einen signifikanten Einfluss der Interface-Variante und eine Interaktion mit der Anzahl der Noten. Für vier und zwölf Noten ist die Fehlerrate für die beiden Varianten des Controllers höher als für die Klaviatur. Dies zeigt, dass die Tonhöhensteuerung über Vier-Finger-Kombinationen, im Sinne der benötigten Zeit, keinen deutlichen Vorteil bei der Tonhöhenauswahl bietet. Im Vergleich zur Klaviatur führt diese Art der Tonhöhenauswahl jedoch zu einer höheren Fehlerrate. Eventuell kann diese durch Übung reduziert werden. Um dies zu untersuchen, sind noch weitere Versuche erforderlich.

Abstract

A gestural interface is realized as part of this Master's thesis, combining the pitch controll and the continious modulation of sound parameters. Pitches are selected by combinations of four fingers and the notes are triggered by an excitation pad. Both provides contious control data, based on the applied pressure. The advantage of controlling the pitch this way is that the fingers can rest on the four keys. Compared to a piano style keyboard, it is not required to position the hand to play another note, but different combinations of fingers need to be pressed at the same time.
The designed controller is evaluated, by investigating whether tones can be played quicker using finger combinations compared to moving the hand to the key position on a keyboard. Which seems to be the most common user interface of digital instruments (Holland et al., 2013a, S. 123).
These two interfaces are difficult to compare, because of various differences. To achieve comparable results, two different versions of the controller are evaluated. Originally, it is required to push the excitation pad on the top of the controller to play the note. In a modified version, the note is played directly when the keys are pressed. Playing all twelve notes of one octave requires finger combinations at the controller and moving the hand to the keyboard key. In order to obtain a corresponding condition, the experiment is additionally carried out with only four notes. This results in a fixed fingering for the keyboard and avoids finger combinations for the controller.
A study is conducted to evaluate the pitch selection of the designed controller. Fingering charts of different pitches are presented as visual stimuli. The subjects are instructed, to play the note as fast and correctly as possible. The times to play the notes and errors are recorded.
Means of subjects average times to play the note and error rates are compared by a two-way ANOVA for repeated measures. The two independent variables are interface type and number of possible notes. In case of detecting a significant difference in the means, interfaces are compared to each other by paired t-tests for twelve and four notes.
For the time to play the note, no significant difference has been detected between the interfaces type and no significant interaction between number of notes and interface type. However the error rates are significantly different for the effect interface type and the interaction between both effects interface type and number of notes. In case of four and twelve notes, the means are significantly higher for both versions of the controller compared to the keyboard.
This indicates, that the four finger combinations provide no advantage in terms of the time to play the note. On the contrary, subjects make more errors compared to the piano keyboard. Probably the error rate could be reduced by training. However, this has not been evaluated in this study. Further evaluations would be necessary.

Inhaltsverzeichnis

Abkürzungsverzeichnis vi

1. Einleitung 1

2. Digitale Musikinstrumente 4
2.1. Beispiele für digitale Musikinstrumente . . . 5
2.1.1. Akai EWI - Ein instrumentenähnlicher Controller . . . 5
2.1.2. Bleauregard - Ein von Blasinstrumenten inspirierter Controller . . . 6
2.1.3. Hyper-Flute - Ein erweitertes Instrument . . . 8
2.1.4. Hand-Controller - Ein alternativer Controller . . . 8
2.2. Musikalische Gesten und Bewegungen . . . 9
2.3. Sensoren . . . 11
2.3.1. Schalter und Taster . . . 11
2.3.2. Drucksensoren . . . 12
2.3.3. Membran-Potentiometer . . . 12
2.3.4. Kapazitive Sensoren . . . 13
2.3.5. Evaluation von Sensoren für musikalische Anwendungen . . . 13
2.4. Mapping - Abbildung der Sensordaten auf Parameter der Klangerzeugung . . . 13
2.4.1. Das Mapping der Tonhöhe . . . 14
2.4.2. Navigation durch den Tonraum . . . 15
2.5. Anforderungen an Controller digitaler Musikinstrumente . . . 15
2.6. Evaluationsmethoden . . . 16
2.6.1. HCI Evaluatiuonsmethoden . . . 17
2.6.2. Unterschiede zu digitalen Musikinstrumenten . . . 19
2.6.3. Evaluation von digitalen Musikinstrumenten . . . 20

3. Methode 23
3.1. Realisierung des Controllers . . . 23
3.1.1. Konzept des Controllers . . . 23
3.1.2. Entwicklungsschritte der Realisierung . . . 24
3.1.3. Mechanische Konstruktion . . . 26
3.1.4. Elektronische Umsetzung . . . 29
3.1.5. Firmware . . . 30
3.1.5.1. Tonhöhensteuerung . . . 31
3.1.5.2. Anregung und Modulation . . . 33
3.1.5.3. Datenübertragung an den PC . . . 33
3.1.6. Klangsynthese . . . 34
3.1.7. Mapping . . . 34
3.1.8. Spielweise . . . 35
3.2. Evaluationsmethode . . . 35
3.2.1. Versuchsaufbau . . . 38
3.2.2. Stimuli und akustisches Feedback . . . 39
3.2.3. Versuchsdurchführung . . . 40
3.2.4. Versuchsteilnehmer . . . 40
3.2.5. Auswertungsmethoden . . . 41

3.2.6. Berücksichtigung der Latenzen 41

4. Ergebnisse **43**
4.1. Ergebnisse des Reaktionstests 43
4.1.1. Reaktionszeiten 43
4.1.2. Fehlerraten 46
4.1.3. Vergleich zwischen den Noten 48
4.2. Ergebnisse des freien Spielens 49
4.3. Ergebnisse des System Usability Scores 51

5. Diskussion **53**
5.1. Reaktionstests 53
5.1.1. Fragebogen zum freien Spielen 54
5.1.2. System Usability Scale 54
5.2. Evaluationsmethode 55
5.3. Realisierung des Controllers 55
5.4. Tonhöhensteuerung über Vier-Finger-Kombinationen 56
5.5. Ausblick 57

Abbildungsverzeichnis **58**

Tabellenverzeichnis **59**

Literatur **60**

A. Versuchsanleitung und Fragebogen **64**

B. Verwendete Materialien **71**

C. Bilder **72**

D. Programmablaufplan **74**

E. Programmcode **75**

Abkürzungsverzeichnis

ADC Analog-to-Digital Converter

ADSR Attack, Decay, Sustain und Release Hüllkurve

ANOVA Analysis Of Variance

ARM Acorn RISC Machine

DAC Digital-to-Analog Converter

DMI Digital Musical Instrument

EVI Electronic Valve Instrument

EWI Electronic Wind Instrument

FSR Force Sensing Resistor

HCI Human Computer Interaction

GUI Graphical User Interface

IDE Integrated Development Environment

K Kleiner Finger

KL Klaviatur

LFO Low-Frequency Oscillator

M Mittelfinger

MSP Max Signal Processing

MIDI Musical Instrument Digital Interface

PCB Printed Circuit Board

R Ringfinger

R2M Ribbon-to-MIDI

RISC Reduced Instruction Set Computing

SD Standardabweichung

SMS Spectral Modeling Synthesis

SPSS Statistical Package for the Social Sciences

SUS System Usability Scale

USB Universal Serial Bus

VO Controller-Variante mit Ventiltasten und Drücken des Excitation Pads

VT	Controller-Variante mit Ventiltasten
Z	Zeigefinger

1. Einleitung

Digitale Musikinstrumente eröffnen einen großen Raum an Gestaltungsmöglichkeiten. Sie fordern die Kreativität von Instrumentenbauern und liefern unzählige Fragestellungen für die wissenschaftliche Forschung.
Die wichtigste Besonderheit von digitalen Musikinstrumenten (DMI) ist die Trennung der Klangberechnung, welche auf einem Computer oder speziellen Chip ausgeführt wird, von der Benutzerschnittstelle (engl. User Interface). Theoretisch ermöglicht diese Trennung eine freie Zuweisung der Steuerparameter auf die Parameter der Klangerzeugung. Dieses Mapping erzeugt eine Kausalität zwischen den Gesten des Musikers und dem Feedback des Instrumentes. Durch die körperliche Verbindung zu dem Controller lässt sich der erzeugte Klang steuern. Da kein unmittelbarer Zugriff auf die Entstehung des Schalls möglich ist, erfassen Sensoren die Gesten. Diese sind an die persönlichen Bedürfnisse des Musikers anpassbar. Künstler können speziell auf sich zugeschnittene Lösungen schaffen, wodurch häufig die Allgemeingültigkeit des Instrumentes verloren geht. Digitale Musikinstrumente bieten sowohl Musikern, als auch Komponisten die Möglichkeit neue Ansätze bei der Umsetzung und Gestaltung von musikalischen Werken zu verfolgen. Auch für das Publikum ändern sich die aufgeführten Inhalte und unter Umständen die gewohnten Gesten und Bewegungen der Musiker.
Die Hauptaufgabe des Musikers besteht beim Spielen eines Melodieinstrumentes darin, durchlaufend unterschiedliche Tonhöhen auszuwählen. Bei mechanischen Musikinstrumenten geben die physikalischen Eigenschaften und die Form des Instrumentes den Bewegungsablauf vor. Die Position des Fingers, an der er die Saite auf das Griffbrett drückt, bestimmt bei der Gitarre die Länge der schwingenden Saite. Eine Änderung der Position des Andruckpunktes, bewirkt eine Veränderung der Tonhöhe. Das Wechseln zu einer anderen Saite bewirkt ebenfalls eine Tonhöhenänderung. Durch das gleichzeitige Anregen von mehreren Saiten ist es möglich Harmonien zu spielen. Hingegen kann die Posaune nur ein Ton nach dem anderen spielen. Jedoch bestimmt auch hier die Position des Zuges die Tonhöhe. Auch bei der Klaviatur, muss bei einem Tonwechsel meist die Hand neu positioniert werden.
Bei der genauen Betrachtung, der Tonhöhensteuerung bei mechanischen Musikinstrumenten zeigt sich, dass die erforderliche Bewegung oft einer Positionierung des Fingers oder der Hand entspricht. Jedoch gibt es auch Ausnahmen. Als ein Beispiel sei hier die Trompete erwähnt, bei der die Tonhöhe durch drei Ventile, die Lippenspannung und den Luftfluss beeinflusst wird. Dabei kann die Tonhöhe durch Drücken des ersten Ventils um einen Ganzton, des zweiten um einen Halbton und durch Drücken des dritten Ventils um eine kleine Terz abgesenkt werden. Bei Kombinationen addieren sich die Intervalle der gedrückten Ventile.
Durch die Trennung des musikalischen Interfaces von der Klangerzeugung gibt es bei der Gestaltung von digitalen Musikinstrumenten zahlreiche Möglichkeiten, die Steuerung der Tonhöhe, einem Bewegungsablauf zuzuweisen. Die Form eines digitalen Musikinstrumentes kann von mechanischen Musikinstrumenten übernommen oder abgeleitet werden, bestehende Instrumente können erweitert oder es können völlig neue Ansätze verfolgt werden.
In dieser Arbeit wird die Steuerung der Tonhöhe eines digitalen Musikinstruments über Vier-Finger-Kombinationen untersucht und praktisch umgesetzt. In einem Versuch wird dieser Ansatz anschließend evaluiert.

Motivation

Digitale Synthesizer bieten meist umfangreiche Einstellmöglichkeiten, um eine große Bandbreite an unterschiedlichen Klängen zu erzeugen. Neben Synthesizern mit eigener Hardware finden native Software-Synthesizer häufig Anwendung. Diese verfügen meist über eine grafische Benutzeroberfläche, welche mit der Computermaus und Tastatur bedient werden kann. Eine angenehmere Steuerung erlaubt ein externer MIDI-Controller. Bei der Verwendung eines MIDI-Keyboards zum Spielen der Töne, wird mit jeder Note die Tonhöhe und die Anschlagstärke übertragen. Neben der Intensität des Klanges, kann diese auch weiteren Parametern zugewiesen werden. Die kontinuierliche Steuerung von Klangparametern bedarf zusätzlicher Controller. Eigene Erfahrungen zeigen, dass es erst durch die kontinuierliche Steuerung mehrerer Klangparameter möglich wird, in die Klangwelten einzutauchen und den Klang durch die haptische Schnittstellen erlebbar werden zu lassen.
Durch Entwicklungsumgebungen für die Klangerzeugung und -verarbeitung, wie Pure Data, Max/-MSP oder SuperCollider, können eigene Klangsynthese-Algorithmen erstellt werden. In Kombination mit geeigneten elektronischen Komponenten können die Daten von Sensoren verwendet werden um die Klangsynthese zu steuern. Häufig entstehen dabei individuelle Lösungen, welche sich für den Künstler als sehr nützlich erweisen, jedoch keine weitere Verbreitung finden.
Das Ziel dieser Masterarbeit ist es, einen Controller zu entwickeln, der neben der Auswahl unterschiedlicher Tonhöhen auch eine kontinuierliche Steuerung der Klangparameter ermöglicht. Sowohl die Tonhöhensteuerung, als auch die Klangmodulation sollen integrale Bestandteile des Controllers sein. Dieser soll eine Alternative zu der weit verbreiteten Klaviatur darstellen. In Kombination mit einer Klangsynthese soll ein flexibel einsetzbares Melodieinstrument entstehen.

Fragestellung

In dieser Arbeit wird die Fragestellung untersucht, ob die Tonhöhensteuerung über Vier-Finger-Kombinationen einen Vorteil gegenüber anderen Arten der Tonhöhensteuerung bietet. Dazu wird im Rahmen der Arbeit ein Controller entwickelt. Bei diesem verweilen die Finger während des Spielens fest auf den Tasten und müssen lediglich gedrückt werden. Dadurch muss die Hand keinen Weg zurück legen, um eine andere Tonhöhe auszuwählen, wie beispielsweise bei der Klaviatur oder dem Griffbrett einer Gitarre. Als Hypothese wird angenommen, dass die Tonhöhenauswahl dadurch schneller erfolgen kann. Außerdem sollte diese Art der Tonhöhensteuerung nicht zu mehr Fehlern bei der Auswahl der Tonhöhe führen und einzelne Töne nicht schwerer zu spielen sein als andere.
Mit einem exemplarischen Mapping und einer einfachen Klangsynthese soll das Instrument hinsichtlich der Usability und der musikalischen Möglichkeiten bewertet werden. Die wesentlichen Fragestellungen dabei sind, ob das Instrument gut steuerbar ist und ob es zu einer Fehlbedienung kommen kann. Bei der Betrachtung der Usability stellt sich auch die Frage, ob die Spielweise leicht erlernbar ist und wie hoch die benötigten motorischen Fähigkeiten eingeschätzt werden. Die Evaluation soll auch Verbesserungsmöglichkeiten aufzeigen und überprüfen, ob das Instrument für Anwender ohne Vorerfahrung einen musikalischen Nutzen bringt.

Aufbau der vorliegenden Arbeit

Zu Beginn dieser Arbeit werden digitale Musikinstrumente allgemein betrachtet, um anschließend ausgewählte Konzepte bestehender Instrumente kurz vorzustellen. Dabei soll der Aspekt der Tonhöhensteuerung im Vordergrund stehen. Im weiteren Verlauf, werden die für diese Arbeit relevanten technologischen Grundlagen geschaffen. Die zur Erkennung von Bewegungsabläufen benötigten Sensoren und mögliche Mapping-Strategien werden dabei beleuchtet. Anschließend werden die Evaluationstechniken der HCI analysiert. In einem weiteren Schritt werden die Unterschiede der Systeme zu den digitalen Musikinstrumenten diskutiert. Zur Veranschaulichung dieser Besonderheiten werden unterschiedliche Evaluationsverfahren digitaler Musikinstrumente vorgestellt.
Der Methodenteil beginnt mit der Beschreibung des im Rahmen der Masterarbeit realisierten Controllers. Es wird das Konzept erläutert und anschließend die mechanische Konstruktion, die Elektronik und die Firmware beschrieben und dokumentiert. Daran schließt sich die Darstellung der verwendeten Evaluationsmethode an. Es werden unter anderem der Versuchsaufbau und die Auswertungsmethoden geschildert. In dem darauf folgenden Kapitel werden die Ergebnisse der durchgeführten Studie präsentiert. Abschließend werden die Ergebnisse, die Untersuchungsmethode sowie Vor- und Nachteile des realisierten Controllers und mögliche Verbesserungen diskutiert.

2. Digitale Musikinstrumente

In diesem Kapitel werden die theoretischen Grundlagen und die unterschiedlichen Kategorien digitaler Musikinstrumente beschrieben. Zu jeder Kategorie wird ein Instrument exemplarisch vorgestellt. Im weiteren Verlauf des Kapitels werden allgemeine Anforderungen an digitale Musikinstrumente besprochen und mögliche Evaluationsmethoden dargestellt.
Werden Musikinstrumente nach dem physikalischen Mechanismus ihrer Klangerzeugung unterschieden, bilden die digitalen Musikinstrumente eine Gruppe, in der Signalprozessoren die Wellenformen des Klanges berechnen. Um diese hörbar zu machen, müssen die digital repräsentierten Schwingungen durch Digital-Analog-Wandler (DAC) in elektrische und durch Lautsprecher in akustische Signale umgewandelt werden. Barbosa et al. (2011) definieren digitale Musikinstrumente allgemein als Musikinstrumente, welche eine Trennung zwischen der Steuerung und der Klangsynthese ermöglichen und bei denen diese Beziehung durch unterschiedliche Mapping-Strategien definierbar ist. Eine direkte Interaktion mit der Klangsynthese ist nur möglich, wenn diese in Echtzeit, d.h. mit einer sehr geringen Latenz, ausgeführt wird. Außerdem müssen die Informationen zur Steuerung ebenfalls als digitale Signale zur Verfügung stehen. Dann ist es möglich diese Steuersignale den Parametern des Klangerzeugungsalgorithmus zuzuweisen und diese interaktiv zu variieren. Die Aktionen des Musikers werden dabei durch Sensoren erfasst und die meist analogen Signale anschließend digitalisiert. Dies kann in einem separaten Steuergerät oder Controller erfolgen. In diesem Fall ist es jedoch erforderlich, dass die digitalen Steuerinformationen an das System, auf dem die Klangsynthese ausgeführt wird, übertragen werden. Dazu ist ein geeignetes Übertragungsprotokoll und eine digitale Schnittstelle mit möglichst geringer Latenz unabdingbar.
Die Controller von digitalen Musikinstrumenten lassen sich nach ihrer Ähnlichkeit zu mechanischen Musikinstrumenten unterscheiden. Miranda und Wanderley (2006) verwenden dazu folgenden Kategorien:

- *Instrumenten ähnliche Controller* emulieren ein mechanisches Instrument. Dadurch lassen sich erlernte Bewegungen weiterhin anwenden. Die Erfahrungen mit mechanischen Instrument lassen sich auf den Controller übertragen.
- *Von klassischen Instrumenten inspirierte Controller* weisen eine Ähnlichkeiten zu mechanischen Musikinstrumenten auf, unterscheiden sich jedoch in wesentlichen Merkmalen.
- *Erweiterte Instrumente* sind mechanische Musikinstrumente, welche z.B. durch Sensoren und eine digitale Klangverarbeitung oder -erzeugung in ihrer Funktionalität erweitert wurden.
- *Alternative Controller* haben in ihrem Design oder ihrer Funktionalität nichts mit mechanischen Instrumenten gemein und können beispielsweise auch Alltagsgenstände sein, welche zu einem Controller umfunktioniert wurden. In diese Kategorie fallen alle Controller, welche völlig neuen Ansätzen folgen und nicht mit mechanischen Instrumenten in Verbindung gebracht werden können.

Sofern nicht die Klangerzeugung betroffen ist, können zur Unterscheidung von digitalen Musikinstrumenten auch Merkmale mechanischer Instrumente angewendet werden. Beispielsweise kann auch hier zwischen Harmonie-, Melodie- oder Percussionsinstrumenten, mono- oder polyphonen Instrumenten oder nach der Art der Benutzung (Blas-, Schlag-, Berührungs- oder Zupfinstrumente), Bestimmung (Konzert-, Spiel-, Lärm-, Rhythmus- oder Effektinstrumente) oder Musikrichtung unterschieden werden. Bei digitalen Musikinstrumenten ist es zusätzlich möglich nach der

Art der Klangsynthese zu unterschieden. Da die Klangerzeugung meist von dem Steuergerät getrennt ist, kann diese in vielen Fällen ausgetauscht werden.

2.1. Beispiele für digitale Musikinstrumente

Zur Veranschaulichung der oben genannten Kategorien, wird in den folgenden Abschnitten zu jeder ein Beispiel vorgestellt. Sofern möglich, richten sich die Beispiele nach einem bestimmten Aspekt der Tonhöhensteuerung, der im Laufe dieser Arbeit wieder aufgegriffen wird.

2.1.1. Akai EWI - Ein instrumentenähnlicher Controller

Der Akai EWI Controller ist ein MIDI-Controller, dessen Form und Spielweise einer Flöte oder Oboe ähnelt. Der Ton wird gespielt, indem in das Mundstück geblasen wird. Der Luftfluss steuert die Amplitude und je nach Einstellung auch weitere Parameter des Klanges. Zur Tonhöhenauswahl befinden sich auf der Vorderseite 13 kapazitive Tasten. Diese werden je nach ausgewähltem Fingersatz unterschiedlich belegt. Zur Auswahl stehen die Fingersätze EWI, Saxophon, Oboe, Flöte und EVI, welcher an den Fingersatz einer Trompete angelehnt ist. Durch die 13 Tasten ergeben sich mehr Kombinationsmöglichkeiten als Tonhöhen die gespielt werden können. Dadurch können einige Tonhöhen über mehrere unterschiedliche Fingersätze ausgewählt werden. Es wird immer der Grundton gespielt, wenn in das Mundstück geblasen wird und keine Taste gedrückt ist.
Auf der Unterseite befinden sich vier Rollen, um die Oktave auszuwählen und zwei Flächen, welche durch Berühren, die momentane Tonhöhe leicht anheben oder absenken. Der Vorteil der kapazitiven Sensoren ist, dass sie auf die Berührung reagieren und keine Kraft aufgewendet werden muss um sie zu betätigen. In das Mundstück ist ein Sensor integriert, welcher auf den Druck der Zähne reagiert und ein Vibrato steuern kann (Akai Professional, 2015).
In der folgenden Tabelle ist der EVI-Fingersatz des Wind-Controllers dargestellt:

Tabelle 1: EVI-Fingersatz nach Akai Professional (2015). Ein gefüllter Kreis bedeutet, dass der Zeigefinger (Z), Mittelfinger (M) oder Ringfinger (R) der entsprechenden Hand gedrückt sein muss.

Links Z M R	Rechts Z M R	Note	MIDI Notennr. einer Oktave
○●●	○○○	C	36
●●●	●●●	C#	37
●●●	●○●	D	38
●●●	○●●	D#	39
●●●	●●○	E	40
●●●	●○○	F	41
●●●	○●○	F#	42
●●●	○○○	G	43
○●●	○●●	G#	44
○●●	●●○	A	45
○●●	●○○	A#	46
○●●	○●○	H	47

Der Fingersatz der rechten Hand, entspricht dem einer Trompete. Da es jedoch nicht möglich ist, über die Lippenspannung eine Naturtonreihe zu spielen, wird zusätzlich der Zeigefinger der linken

Hand benötigt, um alle Halbtonschritte einer Oktave auswählen zu können. Der Mittel- und Ringfinger der linken Hand und die kleinen Finger beider Hände werden nicht zur Tonhöhensteuerung benötigt. Theoretisch würden die vier Finger einer Hand ausreichen, um alle Halbtonschritte einer Oktave auswählen zu können.

Abbildung 1: Akai EWI USB Wind-Controller, Quelle: Akai Professional (2015)

Die Verbindung des Controllers mit dem Computer erfolgt über eine USB-Schnittstelle. Das MIDI-Gerät kann einem Software-Synthesizer oder Sampler ansteuern. Saxophonisten, Flötisten und Oboisten können die erlernten Fingersätze anwenden, wodurch sie ihre Erfahrungen auf den Controller übertragen können. Für Trompetenspieler ist eine Umgewöhnung erforderlich, da anstatt der drei Ventile der Trompete, bei dem Wind-Controller vier Tasten verwendet werden und die Tonhöhe nicht mehr zusätzlich über die Lippenspannung verändert werden kann. Der Controller ist in Abbildung 1 dargestellt.

2.1.2. Bleauregard - Ein von Blasinstrumenten inspirierter Controller

Das Schließen und Öffnen der Löcher von Blasinstrumentes ändert die Länge der schwingenden Luftsäule innerhalb des Instrumentes. Bei Controllern digitaler Musikinstrumente hingegen, bedienen die Finger elektrische Taster, deren Status abgefragt wird und daraufhin die Tonhöhe bestimmt wird. Wenn der Daumen zum Halten des Instrumentes genutzt wird, stehen noch vier Fingern jeder Hand zur Auswahl der Tonhöhe zur Verfügung. Mit jedem weiteren Finger, verdoppelt sich theoretisch die Anzahl der möglichen Tonhöhen. Die Kombinationen aus acht Fingern erlauben die Auswahl von 256 unterschiedlichen Tonhöhen und damit das Spielen über 21 Oktaven. Beauregard (1991) folgt diesem Ansatz und entwickelte einen Wind-Controller mit einem alternativen Fingersatz. Dabei berücksichtigt er, dass ein möglichst großer Tonumfang spielbar ist und es keine bevorzugte Tonart gibt, welche einfacher zu spielen ist als eine andere. Es soll möglichst wenig Intervalle geben, bei denen viele Finger wechseln müssen. Griffe bei denen der kleine und der Mittelfinger gedrückt, der Ringfinger jedoch nicht, sollen vermieden werden, da diese Kombination besonders schwer zu greifen ist.
Der Fingersatz des Bleauregard Wind-Controllers basiert auf Gray-Codes, welche immer einen Hamming-Abstand von eins haben. D.h. die nächst höhere Binärzahl unterscheidet sich nur durch ein Bit. Übertragen auf die Tonhöhensteuerung, muss nur ein Finger geändert werden, um einen Halbton zu erhöhen oder verringern.
Für beide Hände werden identische Fingersätze verwendet. Allerdings übernimmt die rechte Hand die Auswahl der Oktave und die linke die Steuerung der Tonhöhe. Da sich vier Bit Gray-Codes erst nach 16 Zählern wiederholen, jedoch eine Oktave nur zwölf Halbtonschritte zählt, entsteht hier ein Übergang, der ein größeren Hamming-Abstand als eins hat.

Tabelle 2: Fingersatz des Bleauregard Auswahl der entsprechenden Tonhöhe und Oktave basierend auf Gray-Codes nach Beauregard (1991). Der Zeigefinger (Z), Mittelfinger (M), Ringfinger (R) oder (K) kleiner Finger ist bei einem gefüllten Kreis gedrückt.

Links Z M R K	Note	Rechts Z M R K	Oktave
○○○○	C	○○○○	0
○○○●	C#	○○○●	1
○○●●	D	○○●●	2
○●●●	D#	○●●●	3
●●●●	E	●●●●	4
●○●●	F	●○●●	5
●○○●	F#	●○○●	6
●○○○	G	●○○○	7
●●○○	G#	●●○○	8
●●●○	A	●●●○	9
○●●○	H♭	○●●○	10
○○●○	H	○○●○	11

Um dies zu vermeiden entwickelt Beauregard einen Gray-Code, der sich nach 12 Schritten wiederholt und verwendet diesen als Fingersatz. Dieser ist in Tabelle 2 dargestellt. Durch in das Mundstück hineinblasen, wird der ausgewählte Ton gespielt und die Intensität durch den Luftfluss gesteuert. Die Steuerinformationen werden mittels einer speziellen Hardware in MIDI-Befehle umgewandelt. Diese werden an eine externe Klangerzeugung übertragen.

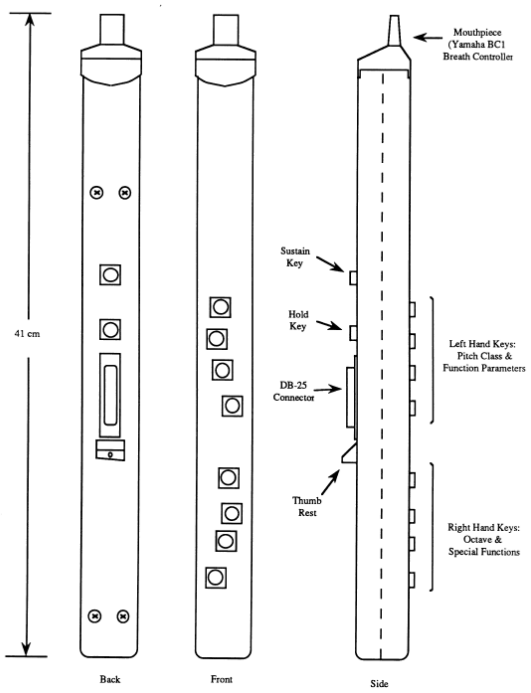

Abbildung 2: Zeichnung des alternativen Wind-Controllers, Quelle: Beauregard (1991)

Da die Oktave durch Fingerkombinationen einer Hand ausgewählt wird, kann über zwölf Oktaven hinweg gespielt werden. Im Gegensatz zum EWI, sind dazu keine zusätzlichen Steuerelemente auf der Rückseite notwendig. Stattdessen, steht für den linken Daumen eine Sustain- und Haltetaste zur Verfügung, wie in Abbildung 2 zu sehen ist. Ein Nachteil ist, dass der Fingersatz anfangs erlernt werden muss und nicht von einem bestehenden Instrument übertragen werden kann. Durch die Verwendung der Gray-Codes, ist eine kleine Sekunde besonders leicht zu spielen. Jedoch müssen für andere Intervalle auch mehrere Finger wechseln.

2.1.3. Hyper-Flute - Ein erweitertes Instrument

Die Hyper-Fulte ist eine klassische Querflöte, welche mit zusätzlichen elektronischen Sensoren versehen ist (Palacio-Quintin, 2003). Die Sensordaten werden in MIDI-Signale umgewandelt und an einen Computer übertragen, um dort den Klang der Querflöte zu beeinflussen. Dazu können unterschiedliche Max/MSP Patches verwendet werden, welche jeweils ein anderes Mapping, der Sensordaten auf die Parameter der Klangverarbeitung vorsehen.
Ein Ultraschall-Abstandssensor ermittelt kontinuierlich den Abstand der Querflöte zum Computer. Drucksensoren sind an den wichtigsten Stellen, an denen die Flöte gehalten wird, angebracht: Unter der linken Hand, welche die Flöte an das Kinn drückt, unter dem rechten Daumen sowie auf der B-Klappe und unter dem linken Daumen. In Reichweite des rechten Daumens befinden sich vier Taster und zwei weitere in der Nähe des Handgelenkes, welche zur Veränderung von Einstellungen genutzt werden oder um ein Stück zu starten oder zu beenden. Bestimmte Neigungswinkel des Instrumentes werden durch Quecksilberschalter erkannt und als MIDI-Noten an den PC übertragen. An den *G#*- und *C#*-Klappen sind Hall-Sensoren angebracht, welche je nachdem wie weit die Klappen geöffnet sind, unterschiedlich stark reagieren.
Durch die Erweiterung der Querflöte mit Sensoren, bleibt die Haptik und Spielweise erhalten. Zusätzlich hat der Musiker die Möglichkeit interaktiv den Klang in einem hohen Maß zu verändern und die Ausdrucksmöglichkeiten zu erweitern. Dies erfordert jedoch zusätzliche motorische Fähigkeiten.

2.1.4. Hand-Controller - Ein alternativer Controller

Der Hand-Controller ist mit einer Vielzahl von unterschiedlichen Sensoren ausgestattet. Ein kleines Gehäuse mit der notwendigen Elektronik ist mit einem Band auf dem Handrücken fixiert. Zur Detektion des Fingerbewegungen werden zwei unterschiedliche Varianten realisiert. Bei der einen, werden die Bewegungen der Finger durch Potentiometer, welche die Form von Trompetenventilen haben, erkannt und bei der anderen durch Membran-Potentiometer. Diese sind auf einer Platte, unter den Fingern angebracht und können die Position und den Druck des Fingers detektieren. Auf einer weiteren Platte befinden sich vier weitere Drucksensoren in Reichweite des Daumens. Durch ein Modul, bestehend aus Beschleunigungssensor, Gyroskop zur Messung der Winkelgeschwindigkeit und einem Magnetometer, zur Messung des Erdmagnetfeldes, können sechs Freiheitsgrade der Handbewegung ermittelt werden (Pardue und Sebastian, 2013).
Die Sensordaten werden seriell über eine USB-Schnittstelle an den Computer übertragen und in einem Max/MSP Patch zur Steuerung der Klangsynthese verwendet. Durch die Potentiometer zur Erkennung der Fingerposition bekommt der Anwender auch haptisches Feedback das wichtig zur genauen und wiederholbaren Steuerung der Parameter ist. Durch die Position des Fingers wird die Tonhöhe über eine Oktave hinweg kontinuierlich gesteuert. Mit jedem Fingers wird die Tonhöhe einer Stimme gesteuert, welche jeweils eine Quinte auseinander liegen. Der Druck der Fingers steuert das Timbre.

Die kontinuierliche Tonhöhensteuerung erschwert das Spielen von diskreten Tonhöhen oder Intervallen. Es ist zu erwarten, dass es viel Übung bedarf, um mit jedem Finger genau eine Tonhöhe zu treffen. Jedoch wird dadurch das Spielen von Glissandi erleichtert.

2.2. Musikalische Gesten und Bewegungen

Dieser Abschnitt behandelt Gesten und Bewegungen beim Spielen von Musikinstrumenten, mit dem Fokus auf die Bewegungsabläufe zur Tonhöhensteuerung. Zunächst werden diese bei mechanischen Instrumenten betrachtet, um daraufhin den Bogen zu digitalen Musikinstrumenten zu schlagen.
Das primäre Ziel der Bewegungen eines Musikers ist es, Klänge zu erzeugen oder zu verändern (Godøy und Leman 2010, S. 5). Meist ist jahrelanges Üben erforderlich um die Bewegungsabläufe zu erlernen und mit konstanter Präzision wiederholen zu können. Die Variabilität in der Bewegung ist bei Anfängern höher als bei erfahrenen Instrumentalisten (Lammers et al., 1996). Das Optimieren der Bewegungsabläufe führt dazu, dass komplexe Passagen flüssig gespielt werden können und ein bestimmter Klang und die Betonung der Melodie möglichst genau reproduziert werden können.
Das neuromuskuläre System hat wesentlich mehr Freiheitsgrade als für die Durchführung einer Bewegung nötig wären (Schubert, 2013). Dadurch ist es fast unmöglich eine Bewegung mehrere Male identisch zu wiederholen. Dies hat bei den meisten Instrumenten einen direkten Einfluss auf den Klang. Andererseits können Bewegungen, welche einen ähnlichen Klang erzeugen, auf unterschiedliche Arten ausgeführt werden. Dies führt dazu, dass Bewegungsabläufe zwischen Musikern nur schwer vergleichbar sind. Abhängig von den individuellen physiologischen Voraussetzungen, entwickelt jeder Musiker die für sich am besten geeigneten Bewegungsabläufe.
Je nach Instrument, sind sehr unterschiedliche Bewegungen erforderlich. Beispielsweise verweilen die Finger beim Spielen von Blasinstrumenten auf den Tasten und werden nur abgesenkt oder angehoben. Hingegen können Pianisten unterschiedlichste Fingersätze verwenden. Bei ihnen ist meist die ganze Hand und der Arm in Bewegung, um mehr Kraft aufzubringen. Beim Spielen von Streichinstrumenten muss der Musiker einerseits die Tonhöhe durch die Position des Fingers auf dem Griffbrett und andererseits die Anregung der Saite mit dem Bogen steuern. Dadurch lässt sich das Timbre und die Feinstruktur des Tones sehr präzise beeinflussen. Die Bewegungen beider Hände müssen zeitlich sehr genau aufeinander abgestimmt werden. Wie bei Streichinstrumenten, kann bei den meisten Instrumenten zwischen Bewegungen zum Anregen und zum Modifizieren des Klangs, beispielsweise der Tonhöhe, unterschieden werden (Godøy und Leman, 2010, S. 25). Bei Saiteninstrumenten übernimmt jede Hand eine der beiden Aufgaben. Bei Blasinstrumenten erfolgt die Anregung durch den Luftstrom, der von der menschlichen Lunge erzeugt wird. Der Luftfluss hat bei dem sogenannten Überblasen einen Einfluss auf die erzeugte Tonhöhe. Dabei wechselt der Grundton je nach stärke des Überblasens zu einem der Obertöne und damit zu einem Ton auf der Naturtonreihe. In der Regel haben die Bewegungen zur Anregung einen starken Einfluss auf den Klang und in manchen Fällen, wie bei den Blasinstrumenten, auch auf die Tonhöhe.
Bei mechanischen Musikinstrumenten kann zwischen einer direkten und indirekten Anregung unterschieden werden. Bei der direkten wird der Klang durch die Bewegung erzeugt, wie bei dem Anschlagen der Gitarrensaiten mit den Fingern. Bei der indirekten Anregung wird dazu ein Gegenstand wie ein Bogen, eine Taste mit Anschlagmechanik oder ein Schlegel verwendet. Bei elektronischen und digitalen Musikinstrumenten ist dies nicht möglich. Die Anregung erfolgt immer indirekt, da nicht direkt auf die Klangentstehung zugegriffen werden kann. Dadurch können sich Bewegungsabläufe ergeben, welche sich stark von denen der mechanischen Musikinstrumenten unterscheiden. Die Situation kann mit der eines Dirigenten verglichen werden. Er verwendet Ges-

ten, um bestimmte Aspekte der Musik zu beeinflussen. Jedoch wird der eigentliche Klang von den Musikern eines Orchesters erzeugt (Godøy und Leman 2010, S. 46). Die entstehenden Klänge von digitalen Musikinstrumenten könnten häufig nicht mit mechanischen Musikinstrumenten erzeugt werden. Andererseits ist es schwer die Dynamik und Komplexität des Klangs mechanischer Instrumente durch digitale nachzubilden.
Neben den Gesten, welche zur Klangerzeugung oder -beeinflussung notwendig sind, gibt es auch Gesten, die zur Kommunikation und zum Ausdruck einer emotionalen Intention dienen, oder eine gewisse Spielweise unterstützen. Diese sind je nach Musiker sehr unterschiedlich, jedoch können manche Besonderheiten beobachtet werden. Bei Holzbläsern wird beispielsweise der Trichter kreisförmig bewegt und Kopf, Schultern oder auch das Instrument angehoben oder abgesenkt (Wanderley et al., 2005). Der Jazztrompeter dagegen bewegt sein Instrument mit gleitender Tonhöhe auf und ab (Godøy und Leman 2010, S. 46).
Bei mechanischen Musikinstrumenten werden die Bewegungen zur Tonhöhenauswahl von dem physikalischen Eigenschaften und der Form des Instrumentes vorgegeben. Eine Übersicht von Levitin et al. (2002) über die gängigen Gesten, welche bei Musikinstrumenten angewendet werden, um die Tonhöhe auszuwählen oder zu verändern, ist in Tabelle 3 dargestellt.
Bei mechanischen Musikinstrumenten ergibt sich die Tonhöhe aus der Länge und Eigenschaft des schwingenden Mediums. Dies kann eine schwingende Luftsäule in einer Röhre, eine Saite, ein Stab, eine Scheibe oder eine Membran sein. Bei der schwingenden Luftsäule wird die Tonhöhe entweder über Öffnen oder Schließen von Löchern an einer bestimmten Position, durch Verändern der Rohrlänge, durch Schieben eines Zuges oder Betätigen von Ventilen gesteuert.

Tabelle 3: Vergleich der unterschiedlichen Bewegungen zur Tonhöhenauswahl

Physik. Eigenschaft	**Instrument oder Medium**	**Geste**
Länge	Länge der schwingenden Luftsäule	Druck zum Schließen oder Öffnen von Löchern
		Druck um Rohrlänge zu verändern
	Schwingende Saite oder Block	Drücken oder Schlagen um Länge zu wählen
Position	Keyboard oder el. Controller	Drücken
Spannung	Membranophon - schwing. Membran	(Anschlagen)
	Blasinstrument	Muskel oder Lippenspannung ändern
	Saiteninstrument	Ändern der Seitenspannung (z.B. ziehen)

Bei der schwingenden Saite bestimmt die Auswahl einer längeren oder kürzeren bzw. weniger oder stärker gespannten Saite die Tonhöhe oder die Saitenlänge wird durch Andrücken verändert. Je weiter dieser Andruckpunkt von dem Körper entfernt ist, desto tiefer ist die Tonhöhe. Bei einer gespannten Membran kann die Spannung geändert oder ein anderer Membrandurchmesser ausgewählt werden. Bei Pedalpauken kann beispielsweise die Spannung der Membran durch ein Pedal geändert werden, dadurch ändert sich die Tonhöhe. Diese wird üblicherweise vor dem Spielen eingestellt und nicht während dessen geändert. Bei schwingenden Blöcken oder Stäben kann das Medium nicht direkt verändert werden, es können hier nur ein Block oder Stab mit einer anderen Länge ausgewählt werden, um die Tonhöhe zu verändern. Die Tonhöhe bei manchen Blechblasinstrumenten kann zusätzlich durch die Änderung der Lippenspannung oder bei Saiteninstrumenten durch Ziehen der Saite beeinflusst werden.
Zur Untersuchung von musikalischen Bewegungsabläufen und zur Steuerung von digitalen Musikinstrumenten ist es notwendig, die Gesten des Musikers zu erfassen. Dazu werden Sensoren verwendet, welche physikalische Größen in elektrische Signale umwandeln. Die zur Tonhöhensteuerung relevanten Sensoren werden in dem folgenden Abschnitt beschrieben.

2.3. Sensoren

Um den Klang digitaler Musikinstrumente ohne einen direkten Zugriff auf die Klangerzeugung interaktiv verändern zu können, werden Sensoren benötigt. Die Bewegungsabläufe, die Körperfunktionen oder Umgebungsparameter können über spezielle Sensoren erfasst und anschließend Parametern der Klangsynthese zugewiesen werden. Die Aktionen des Musikers werden durch die Sensoren detektiert. Da diese nur bestimmte Parameter wie die Position, die Geschwindigkeit oder den Druck messen können, erfassen die Sensoren nur bestimmte Auswirkungen einer Geste und nicht diese als solches (Cadoz, 1988).
Winkler (1995) unterscheidet zwischen Körpersensoren, welche eine Berührung mit dem Körper erfordern und beispielsweise den Winkel, Druck oder Position messen und räumlichen Sensoren, welche durch eine Abtastung des Raumes die Position, den Anstand zu einem Gegenstand oder andere physikalische Eigenschaften der Umgebung messen. Für digitale Musikinstrumente können je nach Anwendung beide Arten von Sensoren verwendet werden. Allerdings bieten die Körpersensoren den Vorteil, dass durch die Berührung ein haptisches Feedback entsteht, welches die Steuerung vereinfacht (Moss und Cunitz, 2005).
Für eine einfache Eingabe oder Erfassung eines Zustandes wie gedrückt oder nicht gedrückt, sind Schalter oder Taster ausreichend. Um jedoch eine physikalischer Größe zur Steuerung der Klangsynthese zu ermitteln, wird ein entsprechender Sensor benötigt.
Kapazitive Berührungssensoren, Schiebe-, Dreh- oder Membran-Potentiometer ermitteln die absolute Position auf einer vorgegebenen Strecke oder Fläche. Größere Abstände, die unabhängig von einer festen Fläche sind, werden über Infrarot, Ultraschall oder Laser-Abstandsensoren gemessen. Zur Erfassung von Bewegungen, muss die Beschleunigung in die jeweilige Richtung bestimmt werden. Dazu dienen Accelerometer. In Kombination mit einem Gyroskop, ist es möglich durch den mehrachsigen Beschleunigungssensor die Lage im Raum zu ermitteln. Dies ist auch über eine Videokamera möglich. Bildverarbeitungsalgorithmen können Marker in dem Bild der Kamera automatisch erkennen und deren Position kontinuierlich verfolgen. Die Position und der Abstand über einer Fläche kann beispielsweise über die projizierte Kapazität detektiert werden.
Die Neigung von Gelenken wird über Biegesensoren oder Drehpotentiometer detektiert. Zur Erfassung von Druck stehen unterschiedliche Sensoren zur Verfügung, wie die Folien-Drucksensoren, Dehnungsmessstreifen oder piezoelektrische Sensoren.
Zur Umwandlung von Wechselgrößen wie Luftschall in eine Spannungen kann ein Mikrofon oder piezoelektrischer Wandler verwendet werden. Zur Erfassung von Schwingung eines ferromagnetischen Gegenstandes dient der elektromagnetische Tonabnehmer.

2.3.1. Schalter und Taster

Schalter und Taster haben die beiden Zustände offen oder geschlossen. Während mechanische Taster nach einer Betätigung zum Ausgangszustand zurückkehren, wechseln Schalter zwischen offen und geschlossen und behalten den jeweiligen Zustand bei.
Taster werden beispielsweise bei MIDI-Keyboards eingesetzt, um zu detektieren, welche Taste gedrückt wurde. Da zusätzlich die Geschwindigkeit (engl. Velocity) des Tastendruckes erfasst werden soll, sind jeweils zwei Taster vorhanden. Bei einem Tastendruck wird der zeitliche Abstand zwischen dem Schließen der beiden Taster ermittelt und der Wert als Velocity mit dem MIDI Note-On Befehl übertragen.
Taster oder Schalter können durch Drücken, Drehen, Schieben, Kippen oder Wippen betätigt werden. Eine Besonderheit stellt der Quecksilberschalter dar, da dieser in einem bestimmten Neigungswinkel schließt. Das elektrisch leitende Quecksilber befindet sich in einem kleinen geschlos-

senen Glasröhrchen mit zwei Kontakten. In einem bestimmten Neigungswinkel berührt es die beiden Kontakte und schließt den Stromkreis. Bei einem Relais schließt der mechanische Kontakt durch einen Elektromagnet. Der Schalter lässt sich durch eine elektrische Spannung steuern. Sobald diese anliegt, wird der ferromagnetische Kontakt angezogen und der Schalter schließt. Sogenannte Halbleiterrelais erfüllen die gleiche Funktion, benötigen jedoch keine mechanische Komponenten. Der Kontakt wird über Halbleiter durch eine Steuerspannung geschlossen und geöffnet. Bei einem Piezoschalter wird diese Steuerspannung durch ein piezoelektrisches Element erzeugt. Diese Schalter sind besonders robust, da sie keine mechanischen Teile enthalten. Bei dauerhaftem Drücken wechselt nach kurzer Zeit der Zustand von geschlossen zurück zu offen, da die Spannung des piezoelektrischen Effekts schnell nachlässt. Dies ist bei der Anwendung dieser Schalter zu berücksichtigen.
Schalter können auch durch Sensoren betätigt werden, wie beispielsweise der Berührungsschalter. Dieser hat ebenfalls den Vorteil, dass er keine mechanischen Elemente benötigt. Das Prinzip basiert auf einem RC-Schwingkreis, dessen Kapazität kontinuierlich gemessen wird. Diese wird durch geringen Abstand oder Kontakt zur Haut beeinflusst. Was durch die zusätzlich benötigte Elektronik detektiert wird.

2.3.2. Drucksensoren

Der Force-Sensing-Resistor (FSR) ist ein druckabhängiger Widerstand und wird sehr häufig für die Realisierung von digitalen Musikinstrumenten verwendet (Marshall et al., 2009). Er besteht aus zwei Folien mit einer leitenden Beschichtung, welche durch Luft getrennt sind. Durch steigenden Druck berühren sich die Folien stärker und der elektrische Widerstand sinkt von etwa 10 MΩ im ungedrückten Zustand bis etwa 1 kΩ bei maximalem Druck. Das Verhältnis zwischen Druck und Widerstand ist nichtlinear, der Widerstand sinkt bei geringem Druck sehr schnell. Mit zunehmendem Druck nimmt die Änderung ab. Dabei reichen 0.2 N, um eine Änderung des Widerstandes zu bewirken. Die maximal erfassbare Kraft liegt bei etwa 20 N, da dort die Sättigung erreicht ist (Interlink Electronics, 2015).
Der durch den Druck steuerbare Widerstand R_{FSR} und einem feste Widerstand R_M bilden einen Spannungsteiler zwischen Masse und der Versorgungsspannung V_{IN}. Dadurch ergibt sich eine dem Druck proportionale Spannung V_{OUT}:

$$V_{OUT} = \frac{R_M V_{IN}}{R_M + R_{FSR}} \tag{1}$$

Durch den geringen Abstand zwischen den Folien, ist der Betätigungsweg vernachlässigbar. Jedoch hat der Sensor eine Ansprechzeit von etwa 3 μs, da die Luft zwischen den Folien entweichen muss. Bei der Konstruktion muss beachtet werden, dass dies möglich ist.

2.3.3. Membran-Potentiometer

Ähnlich wie beim FSR besteht auch ein Membran-Potentiometer aus zwei Folien mit leitender Beschichtung, welche durch Luft getrennt sind. Durch den Fingerdruck berühren sich die beiden Kontaktschichten und es entsteht ein Spannungsteiler. Der Widerstand der Beschichtung nimmt über die Länge des Sensors zu. Dadurch ist die Ausgangsspannung proportional zu der Position, an der die beiden Schichten zusammengedrückt werden. Diese Membran-Potentiometer stellen eine Alternative für Schiebepotentiometer dar.
Als Controller für digitale Musikinstrumente werden Membran-Potentiometer häufig zur kontinuierlichen Tonhöhensteuerung verwendet, wie beim Hand-Controller (siehe Kapitel 2.1.4) oder dem R2M von Doepfer Musikelektronik (2015).

2.3.4. Kapazitive Sensoren

Kapazitive Sensoren messen kontinuierlich die Kapazität, welche durch eine geringe Distanz oder Berührung mit einem leitenden Gegenstand oder dem menschlichen Körper verändert wird. Diese Sensoren können Taster oder Membran-Potentiometer ersetzen oder die Position des Gengenstandes auf einer Fläche detektieren. Der Kondensator besteht aus einer leitenden Fläche, welche zur Detektion dient, und zusätzlicher Elektronik zur Messung des Kapazität. Die Kapazität des Plattenkondensators ergibt sich aus folgender Formel:

$$C = \frac{\varepsilon_r \varepsilon_0 A}{d} \tag{2}$$

Die Gesamtfläche des Sensors A, die elektrische Feldkonstante ε_0 und die Dielektrizitätskonstante ε_r sind dabei konstant. Jedoch ändert sich die Distanz d beispielsweise durch die Annäherung des Fingers an die Sensorfläche und damit die Kapazität. Um diese zu messen, wird an der Sensorfläche eine Spannung angelegt. Diese steigt nach der Ladefunktion des variablen Kondensators und eines zusätzlichen festen Kondensators an. Über die vergangene Zeit, bis ein Schwellenwert erreicht ist, kann die Kapazität ermittelt werden. Meist ist es erforderlich, dass dieser Vorgang mehrmals wiederholt wird, um störungsfreie Werte zu erhalten.

2.3.5. Evaluation von Sensoren für musikalische Anwendungen

Marshall und Wanderley (2006) vergleichen verschiedene Sensoren hinsichtlich ihrer Eignung für musikalische Aufgaben. Dazu werden den Probanden drei Aufgaben gestellt. Sie sollen die Tonhöhen mit dem jeweiligen Sensor steuern, um eine Melodie zu spielen, die Tonhöhe zu modulieren und anschließend beide Aufgaben gleichzeitig umzusetzen. Der Ton der digitalen Klangsynthese wird dabei über einem Druckknopf mit der anderen Hand an- und ausgeschaltet.
Es werden ein Schiebe-, Dreh-, Membran-Potentiometer, ein Drucksensor (FSR) und ein Biegesensor verwendet. Das Membran-Potentiometer erhält für alle Aufgaben die höchste Bewertung. Für die Tonhöhenauswahl ist es gefolgt von dem Schiebe- und Drehpotentiometer, mit etwa gleich hohen Bewertungen im mittleren Bereich. Für die Notenauswahl werden der Druck- und Biegesensor am wenigsten bevorzugt. Anders ist es bei der Modulation der Tonhöhe. Hier liegt der Drucksensor an zweiter Stelle. Bei gleichzeitiger Auswahl und Modulation der Tonhöhe erhalten die beiden Sensoren mit linearer Positionseingabe die höchsten Bewertungen. Als Schlussfolgerung könnte angenommen werden, dass Sensoren, welche eine Positionseingabe über einer linearen Basis ermöglichen, für die Tonhöhensteuerung bevorzugt werden.

2.4. Mapping - Abbildung der Sensordaten auf Parameter der Klangerzeugung

Das Mapping ist die Verbindung zwischen den Steuerparametern, welche von den Aktionen des Musikers abgeleitet werden und den Parametern einer Klangsynthese (Hunt et al., 2000). Die Steuerparameter können auf unterschiedlichste Arten auf die Parameter der Klangsynthese oder -verarbeitung abgebildet werden. Diese Abbildung kann eine spezielle Funktion in einer Komposition übernehmen oder ein fester Bestandteil eines Instrumentes sein.
Die Herausforderung besteht darin einen geeigneten Zusammenhang zwischen den Gesten eines Musikers und den unterschiedlichen Aspekten des erzeugten Klanges zu finden. Dieser Zusammenhang kann entweder durch ein iteratives Verfahren wie beispielsweise maschinelles Lernen oder durch eine explizite Zuweisung hergestellt werden. Nach Levitin et al. (2002) ist das Mapping optimal, wenn dieses intuitiv ist und ein Maximum an Ausdruck bei einem Minimum an

kognitiver Belastung bietet. Die erfassten Gesten sollen auf der physikalischen Ebene eng mit der Intention des Musikers verbunden sein und das Mapping dem Musiker die Möglichkeit geben, seine kognitiven Möglichkeiten auszuschöpfen.
Um das Mapping zu ermöglichen, müssen die Sensordaten in einem geeigneten digitalen Format, mit einer möglichst geringen Latenz vorliegen. Zudem muss die Klangsynthese über Parameter steuerbar sein. Bei der expliziten Zuweisung sind folgende Konfigurationen möglich:

- Eins-zu-eins: Jeder unabhängige Parameter eines Controllers wird einem Parameter der Klangsynthese zugewiesen. Es wird auch als simples Mapping bezeichnet.
- Eins-zu-vielen: Bei dem divergierenden Mapping steuert ein Parameter der Geste mehrere musikalische Parameter gleichzeitig.
- Viele-zu-einem: Hier werden mehrere Gesten zusammengeführt und steuern einen Klangparameter. Dieses Mapping wird daher auch konvergentes Mapping genannt.
- Viele-zu-vielen: Dieses Mapping stellt eine Kombination aus dem divergierenden und konvergenten Mapping dar.

Die von Rovan et al. (1997) vorgeschlagenen Mapping-Strategien werden hier um die Konfiguration von mehreren Parametern der Geste auf mehrere Parameter der Klangsynthese erweitert und mit „viele-zu-vielen“ bezeichnet. Diese Konfiguration ergibt sich, wenn das divergierende und konvergente Mapping kombiniert wird. Beispielsweise kann von einer Geste bei der Druck ausgeübt wird, einerseits der Gesamtdruck und andererseits die Schräge oder Richtung des Druckes erfasst werden. Dadurch sind die beiden Parameter miteinander verknüpft. Diese werden verwendet um jeweils mehrere Parameter der Klangsynthese zu steuern. Hunt und Kirk (1999) verglichen unterschiedliche Mapping-Strategien und schlussfolgern, dass andere Ansätze als eins-zu-eins-Zuweisungen interessanter für die Anwender sind.
Das Mapping kann auch in mehrere Ebenen unterteilt werden. Arfib et al. (2004) verwenden dazu drei Ebenen. Eine übersetzt die Daten der Gesten in Parameter der Gestenwahrnehmung, eine die Parameter der Klangsynthese in Werte der Klangwahrnehmung und eine weitere Ebene verbindet die beiden Wahrnehmungsräume. Den Autoren zufolge bietet diese Struktur mehr Flexibilität und Modularität und erleichtert die Verbindung zwischen den Sensordaten und Klangparametern, da die Verknüpfung auf einer höheren Ebene vorgenommen wird. Außerdem kann zwischen statischem und dynamischen Mapping unterschieden werden. Beim dynamischen Mapping hängt der Steuerparemater der Klangysnthese nicht nur von der momentanen Geste ab, sondern auch von zuvor erfassten Werten. Durch eine Zuweisung, welche sich zufällig ändert oder der stochastische Prozesse zugrunde liegen, erhält das Instrument ein gewisses Eigenleben und der Musiker kann teilweise oder zu bestimmten Zeiten die Kontrolle über sein Instrument verlieren.

2.4.1. Das Mapping der Tonhöhe

Beim Mapping der Tonhöhe muss zwischen der diskreten oder einer kontinuierlichen Tonhöhensteuerung unterschieden werden. Bei der kontinuierlichen ist es möglich mit der Geste der Tonhöhensteuerung auch Vibrato, Glissandi oder andere Effekte der Tonhöhenmodulation zu spielen. Dazu ist die Tonhöhe meist einer festen Wegstrecke zugeordnet und die Position auf der Strecke wird durch einen geeigneten Sensor erfasst, wie beim Hand-Controller in Kapitel 2.1.4 oder dem MIDI Ribbon Controller R2M von Doepfer Musikelektronik (2015). Das Spielen einer Melodie erfordert, dass stets die Position, welcher die benötigte Tonhöhe zugeordnet ist, präzise getroffen wird.

Bei der diskreten Tonhöhensteuerung mit Tasten, sind die vorgegebenen Tonhöhen fest einer Taste zugewiesen, wie bei der Klaviatur, oder mehreren Tasten, wie beim EWI oder Bleauregard (siehe Kapitel 2.1.1 und 2.1.2). Sind die Tonhöhen jeweils Kombinationen von Tasten zugewiesen, verdoppelt sich theoretisch mit jeder weiteren Taste der Tonhöhenumfang des Instrumentes. Wie von Beauregard (1991) beschrieben, könnten mit acht Fingern bereits 256 unterschiedliche Tonhöhen ausgewählt werden. Jedoch geht dadurch die Möglichkeit verloren mehrere Töne gleichzeitig zu spielen. Das Mapping der Tonhöhe kann so gewählt werden, dass es die Übersetzung der Notenschrift in eine Geste vereinfacht. Yunik et al. (1985) entwickelten eine digitale Flöte, bei der jede Note auf einer Notenlinien einer Taste zugewiesen wird. Die fünf Notenlinien und eine Hilfslinie darunter und zwei darüber, sind einer von acht Tasten zugewiesen. Diese werden mit den Fingern beider Hände bedient. Um eine Note zu spielen, welche zwischen zwei Linien liegt, müssen die beiden Tasten der darüber und darunter liegenden Linie gedrückt werden. Zwei zusätzlichen Tasten stehen zu Verfügung, falls die Note mit einem # oder ♭ versehen ist. Diese werden mit dem Daumen der linken Hand gesteuert. Durch dieses Mapping ist nur ein sehr geringes Vorwissen notwendig, um die digitale Flöte zu spielen.
Wichtige Fragen bei der Zuweisung der Fingerkombinationen auf die Tonhöhen sind: Welche Kombinationen sind besonders leicht zu greifen und welche eignen sich besser für bestimmt Tonhöhen? Dies hängt einerseits von der Physiologie der Hand und andererseits von dem Kontext, der Tonart und der Tonfolge ab. Dadurch ist eine allgemeingültige Aussage kaum möglich.

2.4.2. Navigation durch den Tonraum

Alternativ zum direkten Mapping von linearen Positions- oder Tasteninformationen auf die Tonhöhe, ist es möglich mit einen Controller durch einem mehrdimensionalen Tonraum zu navigieren, um dadurch die Tonhöhe zu steuern. Der Controller bildet dabei eine bestimmte musikalisch Logik ab, was das Spielen von Harmonien vereinfachen kann. Beispielsweise beschreiben Gatzsche et al. (2008) wie mit einem SpaceNavigator von 3Dconnexion mit sechs Freiheitsgraden in einem virtuellen Tonraum navigiert werden kann. In diesem Tonraum sind die Töne geometrisch so angeordnet, dass sinnvolle Tonkombinationen nahe beieinander liegen. Bei dem angewendeten Tonraum wird eine Spirale als geometrische Anordnung verwendet. In einer Windung liegen die Terzen einer Tonart benachbart. Der Abstand zu einer benachbarten Windung darüber oder darunter entspricht einem Halbtonschritt. Durch Rotation, Anheben und Absenken kann zu einem beliebigen Ort in dem Tonraum navigiert werden und durch Anhalten, wird der entsprechende Ton gespielt. Durch die Verwendung von Neigungswinkeln soll es auch möglich sein Akkorde zu spielen.

2.5. Anforderungen an Controller digitaler Musikinstrumente

Die Anforderungen an neue Musikinstrumente können teilweise konträr erscheinen. Einerseits sollen sie intuitiv und einfach sein, um die Ziele der Usability zu erfüllen. Andererseits wird eine Komplexität gefordert. Diese soll die musikalischen Fähigkeiten und die Expertise fördern und ein hohes Maß an Expressivität ermöglichen (Marshall et al., 2009). Aus diesem Grund sollte es das Ziel sein, eine Balance zwischen Einfachheit und Komplexität herzustellen (Levitin et al., 2002).
Nach Paine (2013) sollte ein Interface zwei wesentliche Ziele verfolgen: Zum einen soll es dem Musiker ermöglichen die Musik, die er spielt, in einer Weise zu nuancieren, wie es mit einem bestehenden Interface nicht möglich wäre. Zum anderen soll es die Kommunikation mit dem Publikum verbessern. Medeiros et al. (2014) heben hervor, dass ein Interface nicht nur dazu dient, Noten auszuwählen, sondern auch zur Steuerung von klanglichen Eigenschaften. Dabei ist es wichtig,

dass das Instrument einen transparenten Zusammenhang zwischen den Aktionen des Musikers und dem daraus resultierenden Klang herstellt. Dies ist nicht nur für den Musiker, sondern auch für das Publikum ein bedeutendes Kriterium.
In vielen Fällen ist es vorteilhaft, wenn ein Instrument modular gestaltet ist. Dadurch kann die Zuweisung einer Geste auf einen Klangparameter einfach geändert oder der Controller zur Steuerung unterschiedlicher Synthesizer verwendet werden. Allerdings ist es für den Musiker ebenfalls wichtig, dass das Instrument verlässlich auf seine Gesten reagiert und die Einstellungen bestehen bleiben. Denn eine Änderung des Mappings hat zufolge, dass Gesten oder Spielweisen neu erlernt werden müssen.
Bei der Konzeption von digitalen Musikinstrumenten ist es möglich in zwei Richtungen vorzugehen. Die erste Möglichkeit besteht darin von einem gegebenen Klang auszugehen und dessen expressive Möglichkeiten zu erkunden, um anschließend eine zum Klang passende Geste abzuleiten. Die Geste wird dabei durch einen geeigneten Controller erfasst. Die andere Richtung geht von einer Geste aus, um daraus einen passenden Klang zu entwickeln, d.h. die Geste oder ein Aspekt einer Geste wird sonifiziert (Arfib et al., 2004). In manchen Fällen besteht bei der Konzeption bereits eine Idee von der Form oder Funktion des Controllers. Diese gibt dann den Bewegungsablauf vor und anschließend muss der passende Klang entwickelt werden.

2.6. Evaluationsmethoden

Das Ziel einer Evaluation ist es, zu untersuchen ob ein System geeignet erscheint und einen vorgesehenen Zweck erfüllt. Dazu werden Kriterien festgelegt und empirische Daten erhoben. Die Auswertung der Daten soll zeigen, wie das untersuchte System oder der Prozess einzuordnen ist und ob es Verbesserungsmöglichkeiten gibt.
Die Aufgabe von einem Musikinstrument ist je nach Anwendungsbereich und Sichtweise sehr unterschiedlich. Dies kann zu sehr diversen Kriterien führen. Für einen Musiker stehen andere Gesichtspunkte im Vordergrund als für einen Konzertbesucher oder Komponisten. Für das Publikum ist die wahrgenommene Kausalität ein wichtiges Kriterium um ein DMI bewerten zu können. Es sollte bei der Performance eine sinnvolle und wahrnehmbare Verbindungen zwischen den Aktionen des Musikers und dem daraus resultierenden Klang entstehen. Außerdem steht für das Publikum der kommunikative Aspekte der Interaktion im Vordergrund. Für den Musiker spielt es eine große Rolle, wie effektiv die Verbindung zwischen seinen Aktionen und dem Instrument ist. Für ihn ist von Bedeutung, wie hoch die Expressivität und der Umfang der musikalischen Möglichkeiten des Instrumentes sind. Es sollte ihm die nötige Flexibilität und Ausdrucksmöglichkeit bieten, die er für sein Spielen benötigt. Für den Komponist sind eher die Reliabilität und die klanglichen Möglichkeiten wichtige Bewertungskriterien (O'modhrain, 2011).
Für die Evaluation ist es daher wichtig eine Perspektive einzunehmen, um die Kriterien zu definieren. Der Zweck eines Musikinstrumentes ist meist das Spielen selbst. Dabei ist es wichtig, dass das Instrument für den Nutzer motivierend ist und bei dem Spielen ein Fluss (engl. flow) entsteht. Das Instrument zieht dabei den Musiker in den Bann und es stellt sich ein Gefühl der Kontolle und Einklang mit der Tätigkeit ein. Dieser Zustand des „Flows“ entsteht am ehesten, wenn weder eine Unter- noch Überforderung herrscht.
Es ist möglich Musikinstrumente nach deren Bestimmung zu klassifizieren. Die Eingruppierung erfolgt dabei nach der vorgesehenen Funktion des Instrumentes: Zum Beispiel Konzert-, Melodie-, Rhythmus- oder Effektinstrument. Diese Funktion des Instrumentes könnte anschließend zwischen Instrumenten einer Gruppe verglichen und bewertet werden. Beispielsweise dienen Melodieinstrumente dazu unterschiedliche Tonhöhen auszuwählen und den entsprechenden Ton zu spielen. Wird jedoch nur dieser Aspekt isoliert betrachtet, kann noch keine gesamte Bewertung des Instrumentes

erfolgen.
Wie zu Beginn des Kapitels beschrieben, zeichnen sich digitale Musikinstrumente dadurch aus, dass die Steuerung von der Klangerzeugung getrennt ist. Wenn eine Untersuchung des gesamten Instrumentes schwierig ist, können gegebenenfalls einzelne Aspekte des Instrumentes einfacher zu evaluieren sein. Die zur Steuerung notwendigen Gesten, die Sensoren, der Controller oder die Klangerzeugung könnten einzeln untersucht werden. Jedoch lassen sich Verknüpfungen, wie das Mapping, nur schwer von der Geste oder der Klangsynthese trennen.
Für neue Instrumente wäre es möglich diese mit bestehenden zu vergleichen. Die zur Evaluation anzuwendenden Methoden hängen von dem zu untersuchenden Aspekt des Instrumentes und den definierten Bewertungskriterien ab.
Da digitale Musikinstrumente immer einen Computer benötigen, um die Schwingungen der Klänge zu berechnen, ist zur Steuerung eine Benutzerschnittstelle notwendig. Dies führt zu einer Überschneidung mit dem Bereich der Human Computer Interaction (HCI). Deren Ziel ist es, Mensch-Maschine-Schnittstellen zu untersuchen und zu verbessern. In dem folgenden Abschnitt erfolgt eine Einführung in die Evaluationsmethoden der HCI. Jedoch unterscheidet sich der Zweck der zu untersuchenden Systeme von dem der digitalen Musikinstrumente. Diese Unterschiede werden darauffolgend beleuchtet und Beispiele für die Evaluation von digitalen Musikinstrumenten vorgestellt.

2.6.1. HCI Evaluatiuonsmethoden

Für Mensch-Maschine-Schnittstellen wird die Interaktion als ein Prozess definiert, bei dem Informationen zwischen dem Anwender und einer Maschine ausgetauscht werden. Dazu sind Schnittstellen erforderlich, welche mit den Sinnen und den motorischen Fähigkeiten des Benutzers kompatibel sind. Bei Computern wird am häufigsten eine haptische Eingabe und eine visuelle Ausgabe verwendet. Nach Dix (2004) wird ein interaktiver Prozess des Benutzers begonnen um eine bestimmte Aufgabe zu erfüllen. Meist besteht diese darin, den Zustand des Systems zu überwachen und Parameter über Aus- und Eingabegeräte zu ändern (Wanderley, 2002).
Bei der Evaluation von Mensch-Maschine-Schnittstellen steht meist die Usability im Vordergrund. Dabei ist nicht die Benutzerfreundlichkeit, sondern eher die Aufgabenangemessenheit, Anwendbarkeit und Gebrauchstauglichkeit gemeint. Diese wird stark von der menschlichen Leistungsfähigkeit und subjektiven Erwartung beeinflusst. Die Usability ist auch von der Funktionalität und der Aufgabe der technischen Komponente abhängig. Unzureichende Bedienbarkeit resultiert meist aus einer zu technischen Sicht eines Konstrukteurs. Deshalb ist es wichtig, dass unterschiedliche Anwender die Usability überprüfen. Um diese empirisch bewerten zu können, muss sie operationalisiert werden. Bennett (1984) unterteilte Usability in folgende vier Komponenten:

- Leichtigkeit und Schnelligkeit mit der sich die Benutzung erlernen lässt
- Leichtigkeit und Schnelligkeit mit der sich die Benutzerschnittstelle für die Durchführung einer Aufgabe nutzen lässt
- Akzeptanz des Nutzers
- Aufwand den der Nutzer freiwillig betreibt um sich mit dem System besser auszukennen

Für Shneiderman und Plaisant (2005) sind die Fehlerrate des Nutzers, die Merkfähigkeit und die subjektive Zufriedenheit ebenfalls wichtige Kriterien. Im Teil 10 der Norm ISO 9241 werden folgende Kriterien für die Usability von Dialogsystemen genannt:

- Aufgabenangemessenheit
- Selbstbeschreibungsfähigkeit
- Erwartungskonformität
- Steuerbarkeit
- Fehlerrobustheit
- Individualisierbarkeit
- Lernförderlichkeit

Die Bewertung von Eingabegeräten durch eine Betrachtung der physikalischen und mechanischen Eigenschaften, hinsichtlich ihrer Eignung für bestimmte Aufgaben, ist meist nicht ausreichend. Daher ist es erforderlich konkrete Aufgaben zu definieren, für welche die Schnittstelle von Anwendern bewertet werden soll. Dadurch werden die Anwender aktiv in der Evaluationsprozess eingebunden. Um eine Bewertung zu ermöglichen, müssen diese Aufgaben angemessen sein. Die Datenerhebung kann während des Ausführens der Aufgaben oder im Nachhinein erfolgen. Folgende Aufgaben können für die Evaluation von Eingabegeräten durchgeführt werden: Menü- und Textauswahl, Texteingabe und -bearbeitung sowie kontinuierliches Verfolgen eines Objektes. Baecker und Buxton (1987) empfehlen Aufgaben wie die Zielerfassung, mit dem Cursor folgen, Freihandzeichnen, lineare und Kreisbewegungen in vorgegebenen Grenzen. Wetzlinger et al. (2014) stellen den Versuchsteilnehmern die Aufgabe eine E-Mail zu schreiben, einen Kalendereintrag zu erstellen, eine Internetseite aufzurufen und ein Formular auszufüllen, um dadurch Tablets und Laptops miteinander zu vergleichen. Bei der Auswahl der Aufgabe ist zu beachten, dass dadurch messbare Bewertungskriterien geschaffen werden. Wenn mehrere Geräte für eine gegebene Benutzergruppe, Umgebung oder Aufgabe geeignet sind, lässt sich die Usability der Geräte vergleichen. Die Dauer um eine Aufgabe auszuführen ist ein besonders wichtiges Kriterium, da eine geringere Ausführungsdauer zu einer Steigerung der Effizient führt. In diesem Zusammenhang untersucht Fitts (1954) die Dauer, die benötigt wird, um mit einem Stift einen Punkt zu erreichen. Die Abhängigkeit von Distanz und Größe des zu erreichenden Punktes überführt er anschließend in eine Formel, die als Fitts' Gesetz bekannt ist. Dieses Gesetz findet unter anderem bei der Evaluation von Dialogsystemen Anwendung. Hier benötigt der Benutzer eine gewisse Zeit um mit dem Mauszeiger ein Objekt auszuwählen. Viele der Evaluationsmethoden beziehen sich auf Dialogsysteme, da diese weit verbreitete Mensch-Maschine-Schnittstellen sind.
Card et al. (1978) vergleichen die Performance der Textauswahl mit unterschiedlichen Eingabegeräten. Dazu ermitteln sie die Fehlerrate und Dauer, um zu einer vorgegebenen Textstelle zu navigieren und diese auszuwählen. Es werden zur Auswertung die gemessenen Werte der unterschiedlichen Eingabegeräte verglichen. Da dabei signifikante Unterschiede auftreten, könnte von den Ergebnissen abgeleitet werden, dass sich einzelne Eingabegräte mehr oder weniger gut für diese Aufgabe eignen.
Wie bei der Messung der Ausführungsdauer können die Daten der Aktionen und die jeweiligen Zeiten automatisch gespeichert werden. Die Daten werden während der Benutzung erhoben, ohne dass der Anwender daran beteiligt ist. Weitere Möglichkeiten diese Daten zu erheben, sind beispielsweise durch die menschliche Beobachtung, Videoaufzeichnung, Schreiben von Logfiles, Eye-Tracking oder durch die Messung physiologischer Daten. Allgemein wird die Aktivität des Anwenders erfasst und die Daten werden anschließend ausgewertet. Es können beispielsweise die Bearbeitungszeit, Nutzungsstatistiken einzelner Funktionen oder Fehlerhäufigkeiten analysiert

werden. Allerdings ist es nicht möglich Informationen über die subjektive Erfahrung, Erwartung, emotionale Reaktion oder Verbesserungsvorschläge des Anwenders zu erhalten. Dazu ist eine Befragung des Benutzers notwendig. Bei der Befragung oder durch lautes Denken nimmt der Anwender eine bewertende Rolle ein und trägt aktiv zur Datenerhebung bei. Beim lauten Denken wird der Anwender gebeten während der Ausführung von Aufgaben seine Gedanken laut auszusprechen. Dadurch gewährt er Einblicke in seine Gedanken, Wahrnehmungen, Empfindungen und Absichten. Jedoch ist es umstritten, ob die kognitiven Prozesse hinreichend und vollständig verbalisiert werden können oder dadurch beeinflusst werden (Mey und Mruck, 2010, S. 486).
Ein Interview wird durchgeführt, um ein subjektives Urteil und die Meinung des Anwenders einzuholen. Dadurch können Informationen über dessen persönliche Relevanzsystem gesammelt werden. Bei einem offenen Interview soll der Befragte möglichst frei erzählen, um selbst anzugeben, was für ihn persönlich relevant ist. Hingegen folgt das halb strukturierte Interview einem Leitfaden, um dem Gespräch eine Struktur zu verleihen und die Themengebiete im Vorhinein festzulegen. Der Anwender schildert in der Retrospektive seine Erfahrungen oder subjektive Beurteilung des Systems oder seines Nutzungsverhaltens. Die gestellten Fragen zielen dabei auf bestimmte Aspekte des zu untersuchenden Systems ab. Beispielsweise wird bei der Critical Incident Technic der Befragte dazu aufgefordert, wahllos besonders gute und besonders schlechte Erfahrungen während der Benutzung des Gerätes zu beschreiben, um Defizite aufzudecken (Backhaus, 2009).
Ein Beispiel für quantitative Usablility Evaluationsmethoden ist die System Usability Scale (SUS) nach Jordan und B. Thomas, Ian Lyall McClelland (1996). Bei diesem Verfahren beurteilt der Anwender zehn Aussagen, welche abwechselnd positiv oder negativ formuliert sind, auf einer Likert-Skala. Anhand der Antworten wird ein SUS-Score errechnet, welcher zwischen 0 und 100 liegen kann. Dieser Score gibt nur einen allgemeinen Überblick über die Usability und lässt keine eindeutigen Rückschluss auf die Ursache eines hohen oder niedrigen Wert zu.
Zur Übertragung von HCI-Evaluationsmethoden auf DMI, ist es notwendig die Aufgaben, entsprechend der zu untersuchenden Funktion, anzupassen und eine geeignete Art der Datenerhebung zu wählen. Zwischen den zu untersuchenden Systemen gibt es jedoch grundlegende Unterschiede, welche bei der Evaluation zu berücksichtigen sind. Diese werden in dem folgenden Kapitel beschrieben.

2.6.2. Unterschiede zu digitalen Musikinstrumenten

Digitale Musikinstrumente dienen meist einem anderen Zweck, als die im Bereich der HCI untersuchten Benutzerschnittstellen. Daher ist es wichtig die wesentlichen Unterschiede herauszuarbeiten, um anschließend einschätzen zu können, ob die Methoden der HCI auch für DMI angewendet werden können. Folgende Unterschiede werden von Hunt und Kirk (2000) als besonders wichtig angesehen:

- Es gibt keine feste Ordnung und Reihenfolge von Dialogen.
- Der Mensch übernimmt die Kontrolle und der Computer reagiert entsprechend.
- Es gibt nicht einen eingeschränkten Satz an erlaubten Operationen, sondern eine gewisse Menge von kontinuierlichen Steuermöglichkeiten.
- Die Bewegungen des Benutzers verursachen eine sofortige Reaktion.
- Anstatt der geordneten Übertragung von Informationen liegt das Hauptziel in der Kontrolle über das gesamte System.

- Die Steuerung erfolgt mit einem physikalischen, multi-parametrischen Steuergerät und muss erlernt werden, bis sich Bewegungen automatisiert haben.
- Viel Üben führt zu einer verstärkten Kontrolle über das System und dadurch zu mehr Kompetenz in der Anwendung.
- Wenn der Benutzer mit dem System vertraut ist, kann er auch andere kognitive Aktivitäten während der Anwendung durchführen.

Außerdem erfüllen musikalische Controller andere primäre Aufgaben als gängige Eingabegeräte wie eine Computermaus oder Tastatur, welche meist dazu dienen eine Auswahl zu treffen oder Zeichen einzugeben. Bei Musikinstrumenten sind die Anwender voll und ganz für die Aktionen verantwortlich und steuern viele Parameter eines komplexen musikalischen Systems gleichzeitig. Sie erhalten dabei kontinuierliches Feedback und der Einfluss jeder einzelnen Aktionen wird von Moment zu Moment mit dem ganzen Körper erlebt (Hunt und Kirk, 2000). Dies kann dazu führen, dass das gesamte Instrument nur schwer evaluiert werden kann. Meist erfüllen jedoch einzelne Teile des Systems bestimmte Funktionen. Dadurch können diese Fragmente über vorgegebene Aufgaben hinsichtlich ihres Nutzens und ihrer Bedienbarkeit evaluiert werden.

2.6.3. Evaluation von digitalen Musikinstrumenten

In Anlehnung an die HCI Forschung, empfiehlt Wanderley (2002) zur Evaluation von Controllern digitalen Musikinstrumenten den Versuchspersonen musikalische Aufgaben zu stellen. Dadurch lässt sich ermitteln, welcher Controller für diese Aufgabe, die beste Performance liefert. Diese Art der formalen Evaluation wird als ein erster Schritt betrachtet, um die Möglichkeiten und Grenzen eines Controllers besser zu verstehen. Denn meist ist es nicht schwer den gesamten kreativen Prozess des Musizierens zu evaluieren. Da dieser für jeden Musiker sehr unterschiedlich sein kann und dadurch schwer zu vergleichen ist.
Bei der Evaluation ist es wichtig aus der Aufgabe messbare Performance-Parameter abzuleiten. Wenn ein erfahrener Musiker ein neues Instrument ausprobiert folgt er meist nicht quantitativen Kriterien (Wanderley, 2002). Daher sollten diese Performance-Parameter im Gegensatz zur HCI-Forschung nicht quantitativ erfasst werden. Im musikalischen Kontext ist die Bewertung nicht von der subjektiven Wahrnehmung des Künstlers und dessen persönlichen und ästhetischen Kriterien trennbar.
Die von Wanderley (2002) vorgeschlagenen Methoden finden unterschiedlichste Anwendungen. Beispielsweise werden sie von Marshall et al. (2009) zur Evaluation von unterschiedlichen Strategien zur Tonhöhenmodulation eingesetzt. Die musikalische Aufgabe besteht darin die Tonhöhe jeweils gleitend, drückend und rollend zu modulieren. Zur Evaluation werden sowohl objektive als auch subjektive Kriterien angewendet. Zu den objektiven zählt die Stabilität der Vibratogeschwindigkeit. Zu den subjektiven zählen die Bewertung der Bedienung, die Bevorzugung der schnellen oder langsamen Geschwindigkeit, die verwendete Hand und die gesamten Präferenz. Die Versuchsteilnehmer bewerten jeweils die Art der Modulation auf einer Skala von eins bis neun. Der Versuch wird mit Violinisten und Pianisten durchgeführt, um einen Einfluss der Vorerfahrung zu berücksichtigten. Für das musikalische Können (Pianisten im Vergleich zu Violinisten) konnte kein signifikanter Unterschied festgestellt werden.
In dem ersten Experiment wird ein hoher Einfluss der Vibratogeschwindigkeit auf die Stabilität festgestellt. Dieser soll in einem weiteren Experiment untersucht werden. Die Probanden bekommen die Aufgabe gestellt, ein vorgegebenes Vibrato mit unterschiedlichen Geschwindigkeiten, jeweils zeitgleich nachzuspielen. Das objektive Kriterium bilden die Stabilität, die Genauigkeit

und die Vibratotiefe. Die Ergebnisse zeigen, dass die Geschwindigkeit den größten Einfluss auf die objektiven Kriterien hat. Für Modulationsraten größer als 3 Hz erzielt die gleitenden Modulation eine signifikant höhere Stabilität. Während für die drückende und rollende Vibratomethode, kein eindeutiger Unterschied in der Stabilität, Genauigkeit und Modulationstiefe aufgetreten ist.
Auch wenn in dem zweiten Experiment keine subjektiven Bewertungen erfasst wird, zeigt der erste Versuch, dass die objektiven Daten keinen Unterschied zwischen der rollenden und drückenden Methode aufzeigen. Jedoch weisen die subjektiv erhobenen Daten auf Unterschiede hin. Dies zeigt, dass sehr hilfreich ist subjektive Daten zu erheben.
Um zu überprüfen ob sich Evaluationsmethoden der HCI für musikalische Interfaces anwenden lassen, führen Kiefer et al. eine Fallstudie durch. Dabei wird untersucht, wie nützlich die Wiimote als musikalischer Controller ist. Die musikalischen Aufgaben, welche den Probanden gestellt werden, ist das Triggern von Klängen über Schlagbewegungen, das kontinuierliche Steuern eines Parameters und die Gestensteuerung, bei der bestimmte Bewegungen erkannt werden. Die interessantesten Ergebnisse werden dabei durch das durchgeführte Interview erzielt. Besonders bedeutend sind dabei die vorgeschlagenen Verbesserungen und alternativen musikalischen Anwendungsmöglichkeiten des Wii-Controllers. Allerdings gibt das anschließende Interview auch nur eingeschränkte Einblicke in die Erfahrungen während der Bedienung des Gerätes.
Vertegaal und Eaglestone (1996) verglichen die Navigation im vierdimensionalen Timbre-Raum mit der Computermaus, einem relativ und absoluten steuerndem Joystick und dem Dataglove. Die Aufgabe besteht darin einen vorgegeben Punkt in dem Timbre-Raum möglichst genau zu treffen. Zur Auswertung wird die Dauer um den Punkt zu erreichen und der Abstand, zu dem vorgegebenen Punkt, ermittelt. Anstatt gemessene Werte zu betrachten, vergleichen Wanderley et al. (2000) die subjektiven Bewertungen von Versuchsteilnehmern. Bei dem Experiment sollte ein Vibrato mit einem Drucksensor, Positions- und Drehsensor erzeugt und anschließend bewertet werden.
Ghamsari et al. (2013) versuchen musikalische Aufgaben und Improvisation zur Evaluation eines DMI zu kombinieren, um eine möglichst realistische und musikalische Situation zu schaffen. Dabei wird untersucht, wie Musiker sich mit einem neuen DMI ausdrücken und welchen Einfluss das Mapping darauf hat. An der Evaluation waren vier professionelle Musiker, mit langjähriger Improvisationserfahrung, unter andrem mit DMI, beteiligt. Es werden zwei unterschiedliche Mappings für das DMI „The Ballagumi“ verglichen. Bei dem einen werden die Sensordaten direkt verwendet und bei dem anderen die Beschleunigung, d.h. die momentane Änderung. Der Controller des DMIs besteht aus einer Silikonform mit integrierten optischen Biege- und Drucksensoren und ist frei formbar. Dadurch wird keine Spielweise vorgegeben. Innerhalb von zwei Sitzungen wird je ein Mapping untersucht. Zu Beginn besteht die Aufgabe darin drei Ausschnitte aus zuvor aufgenommenen Stücken nachzuspielen. Diese Stücke sind nach den Vorgaben von Butler (2008) komponiert und beziehen sich auf unterschiedliche Spieltechniken digitaler Musikinstrumente. Die Musiker werden dazu aufgefordert, die Stücke nach Gehör nachzuspielen. Anschließend wird ein kurzes Interview durchgeführt und nach einer kurzen Lernphase folgt die Improvisationssession und ein weiteres Interview. Die Antworten der Befragungen werden qualitativ analysiert. Es stellt sich dabei heraus, dass die Lernkurve durch die zeitlich begrenzten Sitzungen nur schwer bewertet werden kann. Außerdem erschwert der Wechsel des Mappings das Erlernen der neuen Spielweise. Die Teilnehmer bevorzugen mehrheitlich das direkte Mapping und empfinden dieses als intuitiver. Die Latenz des Controllers erschwert perkussives Spielen. Die Methode erweist sich nach Meinung der Autoren als nützlich, da sich die Teilnehmer auf das Musizieren konzentrieren können und dadurch mehr Feedback in den Interviews geben, als wenn sie nur Aufgaben erfüllen sollten.

Diese Beispiele zeigen, dass sich Methoden der HCI für die Evaluation einzelner Aspekte musikalische Controller anwenden lassen. Die Ausführung musikalischer Aufgaben liefert vergleichbare Ergebnisse, jedoch lassen sie keine Rückschlüsse auf das musikalische Potential des gesamten Controllers ziehen. Stowell et al. (2009) kritisieren, dass durch die Reduktion, auf maximal vereinfachte musikalische Aufgaben, die Authentizität der Interaktion und der kreative Aspekt des Musizierens verloren gehen.
Stowell et al. (2009) wenden die Diskursanalyse als qualitative und den Turing-Test als quantitative Evaluationsmethode an. In der Diskursanalyse soll das Konzept der musikalischen Schnittstellen vom Benutzer bewertet werden. Die Probanden machen sich dazu in Einzelsitzungen anfangs frei mit dem Interface vertraut und wenden diese auf ihre eigene Art und Weise an. Anschließend folgt ein geführtes Erkunden und ein halb strukturiertes Interview. Die Aussagen der Einzelsitzungen werden transkribiert und dazu verwendet erste Eindrücke als freie Assoziationen festzuhalten. Außerdem werden die Texte dazu verwendet jedes Objekt und jeden Akteur des Diskurses aufzulisten und einzuordnen. Anschließend wird die beschriebene Umgebung rekonstruiert und der Kontext untersucht. Durch eine abschließende geleitete Gruppendiskussion konnten die zuvor gefundenen Kategorien, Benennungen und Vergleiche besprochen werden. Die Autoren kommen zu dem Fazit, dass die Diskursanalyse hilfreich ist um das System im Bezug auf sich selbst und andere Objekte einzuordnen.
Wenn das Ziel der Evaluation nicht darin besteht, die Expressivität eines digitalen Musikinstrumentes, sondern einzelne Aspekte der Steuerung zu untersuchen, liefert die aufgabenbezogene Herangehensweise aufschlussreiche Ergebnisse. Diese Methode wird zur Bewertung der Tonhöhensteuerung über Vier-Finger-Kombinationen angewendet.

3. Methode

Das Thema dieser Arbeit ist die Entwicklung und Evaluation eines Controllers zur Tonhöhensteuerung über Vier-Finger-Kombinationen. Dabei sollen die im vorangegangen theoretischen Grundlagen berücksichtigt werden. In diesem Kapitel wird der Entwicklungsprozess und der resultierende Prototyp beschrieben. Daran schließt sich die Erläuterung der Evaluationsmethode an. Diese umfasst unter anderem den Versuchsaufbau und -ablauf, die Stimuli und die Auswertungsmethode.

3.1. Realisierung des Controllers

Dieser Abschnitt beschäftigt sich mit dem Entwicklungsprozess und den einzelnen Elementen des realisierten Controllers. Er besteht aus unterschiedlichen Hardware- und Software-Komponenten und dient der Steuerung einer digitalen Klangsynthese. Das Konzepte und die Entwicklungsschritte des flexibel einsetzbaren Controllers werden erläutert. In den darauffolgenden Abschnitten werden die mechanische Konstruktion, Elektronik und Firmware dokumentiert. Als ein Beispiel wird die mit Pure Data realisierte Klangsynthese und das Mapping beschrieben. Über die MIDI-Schnittstelle ist es möglich auch andere Synthesizer anzusprechen.

3.1.1. Konzept des Controllers

Die Grundidee des musikalischen Interfaces ist es, die Tonhöhenauswahl über Tastenkombinationen zu ermöglichen. Dadurch verdoppelt sich der mögliche Tonhöhenumfang mit jeder weiteren Taste. Die Tonhöhensteuerung soll nur mit einer Hand erfolgen, sodass die Anregung und die Modulation des Klangs von der anderen Hand übernommen werden können. Durch Kombinationen aus den vier Fingern einer Hand können bereits 15 unterschiedliche Tonhöhen ausgewählt werden, wenn der Zustand, in dem keine Taste gedrückt ist, unberücksichtigt bleibt. Dies überschreitet bereits den Tonumfang einer Oktave. Die Zuweisung zwischen den Fingerkombinationen und Tonhöhen kann auf unterschiedliche Arten erfolgen. Teil des Konzeptes ist es, die Fingerkombination als Binärzahl zu interpretieren und die Zahlen den Halbtonschritten zuzuweisen. Dadurch besteht ein logischer Zusammenhang zwischen der Fingerkombination und der Tonhöhe. Jedem Finger wird die Wertigkeit einer Zweierpotenz zugeordnet; dem Zeigefinger 2^0, dem Mittelfinger 2^1, dem Ringfinger 2^2 und dem kleinen Finger 2^3. Ist der Finger gedrückt, wird das entsprechende Bit gesetzt. Ausgehend von dem Grundton $C2$, wenn nur der Zeigefinger gedrückt ist (0001), kann die Tonhöhe in Halbtonschritten ausgewählt werden. Die Fingerkombination 0110 entspricht beispielsweise einem F (siehe Tabelle 4).

Tabelle 4: Mapping der Fingerkombinationen auf die Tonhöhe. Bei einer Eins ist der kleine Finger (K), Ringfinger (R), Mittelfinger (M) oder Zeigefinger (Z) gedrückt.

Fingersatz K R M Z	Note	MIDI Notennr. der ersten Oktave
0 0 0 1	C	36
0 0 1 0	C#	37
0 0 1 1	D	38
0 1 0 0	D#	39
0 1 0 1	E	40
0 1 1 0	F	41
0 1 1 1	F#	42
1 0 0 0	G	43
1 0 0 1	G#	44
1 0 1 0	A	45
1 0 1 1	A#	46
1 1 0 0	H	47
1 1 0 1	C	48
1 1 1 0	C#	49
1 1 1 1	D	50

Die vier Finger bleiben dauerhaft auf den Tasten. Zur Tonhöhenauswahl drücken sie die entsprechende Taste. Ein Umgreifen ist dabei nicht notwendig. Während dem Spielen ist es nicht notwendig, den Blick auf die Tasten zu richten, um eine neue Position mit den Fingern zu finden. Da zu einem Zeitpunkt immer nur eine Tonhöhe ausgewählt werden kann, ist der Controller monophon. Dieses Konzept scheint eine mögliche Alternative zu der Position-Tonhöhen-Zuordnung zu sein, welche bei vielen anderen Instrumenten vorzufinden ist. Ob dieses Art der Tonhöhensteuerung einen Vor- oder Nachteil, gegenüber bereits verbreiteter Arten der Tonhöhenauswahl darstellt, soll mit dem realisierten Interface untersucht werden.
Durch dieses Konzept der Tonhöhenauswahl ergeben sich für die Gestaltung des Controllers eine Vielzahl an Vorgaben. Der folgende Abschnitt geht auf die einzelnen Schritte der Entwicklung des Controllers ein.

3.1.2. Entwicklungsschritte der Realisierung

Die Grundidee des Controllers, Fingerkombinationen in Form von Binär-Codes zu verwenden, stammt von Henrik von Coler, von dem Fachgebiet Audiokommunikation, und wird für die Umsetzung des Controllers aufgegriffen. Die mechanische Konstruktion wurde von Dr. Thomas Lerch vorgenommen, der als Restaurator für Blas- und Schlaginstrumente am Staatlichen Institut für Musikforschung tätig ist. Er lässt seine langjährigen Erfahrungen als Instrumentenbauer in die praktische Umsetzung mit einfließen. Während der gesamten Entwicklungsphase sind Henrik von Coler und Dr. Lerch beratend tätig und an Entscheidungen zur Realisierung des Controllers beteiligt. Die elektronische und programmiertechnische Realisierung ist Teil dieser Masterarbeit.
Zum Drücken der Tasten eignet sich der Kraftgriff als Geste. Dazu werden die Tasten an einem Rohrabschnitt angebracht. Zur Detektion des Tastendruckes werden anstatt einfacher Schalter Drucksensoren verwendet, um durch die Druckintensität ebenfalls als Parameter zur Steuerung der Klangsynthese gewinnen zu können. Diese Art der Aftertouch-Steuerung wird auch bei zahlreichen anderen elektronischen Tasteninstrumenten angewendet.

Es wird ein kompressibles Material benötigt, damit die Tasten durch den Druck der Finger einen Weg zurücklegen und die Druckintensität dadurch besser gesteuert werden kann. Ein Vergleich zwischen unterschiedlichen Materialien zeigt, dass sich sehr weiches Silikon gut eignet. Von dem Hersteller wird das verwendete Material mit „Shore 00 super weich“ gekennzeichnet (Troll Factory Rainer Habekost, 2015). Die beiden Komponenten, Basis und Vernetzer, werden im Verhältnis 1:1 gemischt. Anschließend lässt sich das Silikon zum Aushärten in eine beliebige Form gießen. Damit die Silikonscheiben die aktive Fläche der FSR-Senoren abdecken, wird ein unten geschlossener Zylinder mit einem Durchmesser von 18 mm und einer Höhe von 9 mm verwendet.
Um die Tasten zu befestigen und den Laufweg einstellen zu können, werden vier Zylinder aus Messing in 20 mm Abständen mit der Rohrwand verschraubt. Diese sind durch ein Gewinde und eine Mutter höhenverstellbar. In dem Zylinder können sich die Tasten auf und ab bewegen. Die Tasten wurden von Dr. Lerch so konstruiert, dass sie aus zwei verschraubbaren Teilen bestehen, die am Anfang und Ende über runde Flächen verfügen. Auf die obere drückt jeweils ein Finger und durch die untere wird der Druck auf die Silikonscheiben weitergegeben. Außerdem kann die Ruheposition durch die höhenverstellbaren Zylinder so eingestellt werden, dass gerade noch kein Druck auf die Sensoren ausgeübt wird. Die Tasten ähneln optisch den Ventilen einer Trompete, außer dass sie nur aus einem offenen Bolzen bestehen. Aufgrund der Ähnlichkeit werden diese Tasten im folgenden Text als Ventiltasten bezeichnet.
Die Sensoren werden zwischen zwei Holzplatten fixiert und befinden sich auf einer Fläche im inneren des Rohres. Die obere Holzplatte hat runde Aussparungen und abgerundeten Kanten, damit die Silikonscheiben auf Position der Sensoren bleiben und sich durch den Druck ausdehnen können.
Durch den Kraftgriff liegt der Daumen in Opposition mit den anderen Fingern. Neben dem Halten des Controllers wird er verwendet, um die Oktave über Druckschalter auszuwählen. Dazu werden drei Piezo-Schalter auf der gegenüberliegenden Seite der vier Ventiltasten angebracht.
Durch die Ventil- und Oktavtasten kann nur die Tonhöhe gesteuert werden. Zusätzlich ist ein weiterer Mechanismus notwendig, um die Note zu spielen und den Klang zu modulieren. Dazu wurden unterschiedlichste Konzepte entwickelt und diskutiert.

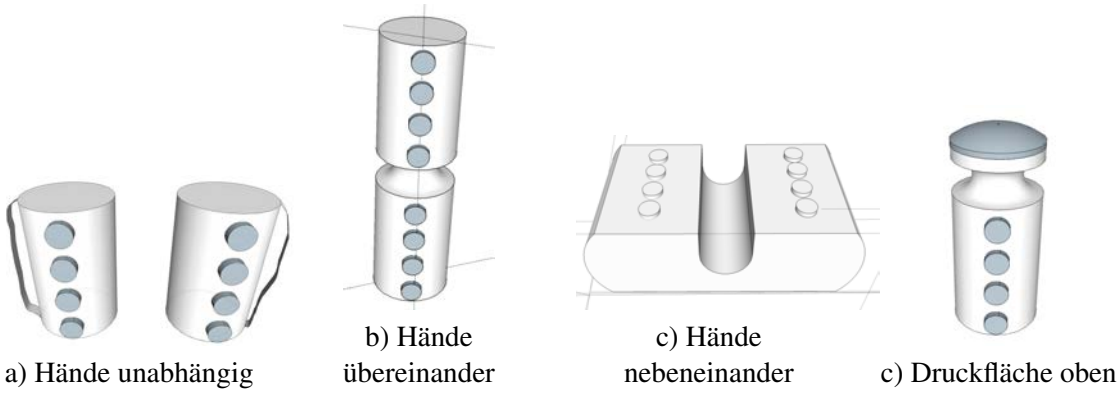

Abbildung 3: Skizzen der unterschiedlichen Konzepte

Die möglichen Konfigurationen sind in Abbildung 3 dargestellt. Es können entweder zwei unabhängige Controller für jede Hand realisiert werden a), von denen einer die Tonhöhe und der andere den Klang moduliert. Alternativ können die Hände übereinander b) oder nebeneinander c) an einem Controller angeordnet werden. Die Variante mit einer Druckfläche am oberen Rohrende d), zum Auslösen und Modulieren des Tones scheint am geeignetsten zu sein. Das Kopfstück wird so

konstruiert, dass es eine Verjüngung hat, die zum Halten des Controllers dient. Am oberen Ende ist eine bewegliche Holzscheibe untergebracht. Diese wird an vier Punkten von Silikonscheiben gestützt, welche den Druck auf darunterliegende FSR-Sensoren übertragen. Dieser Mechanismus, der durch Druck sowohl zum Auslösen, als auch zur Modulation des Klanges dient, kann als Excitation Pad bezeichnet, das ähnlich wie bei vielen mechanischen Musikinstrumenten die Funktion der Anregung übernimmt.
Die Elektronik wird ebenfalls auf der Fläche im Inneren des Rohres untergebracht. Am Unteren Ende schließt ein Fuß aus Holz das Rohr ab und dient dazu den Controllers abzustellen.

3.1.3. Mechanische Konstruktion

Die Tasten zur Tonhöhensteuerung und Oktavauswahl sind gegenüberliegend an einem transparentem Acrylrohr untergebracht. Das Rohr hat eine Länge von 180 mm, einen Durchmesser von 70 mm und eine Wandstärke von 3 mm. In dem Rohr ist mittig eine Acrylplatte eingezogen, welche auf beiden Seiten des Rohres durch Schrauben befestigt ist. Sie bildet im Inneren des Rohres einen Querschnitt und hat eine Stärke von 8 mm. Durch das Lösen der Schrauben kann die Platte herausgenommen werden, um beispielsweise Änderungen an der Elektronik vorzunehmen. In der Abbildung 4 ist die Zwischenebene herausgenommen und auf der linken Seite zu sehen, rechts das Acrylrohr.

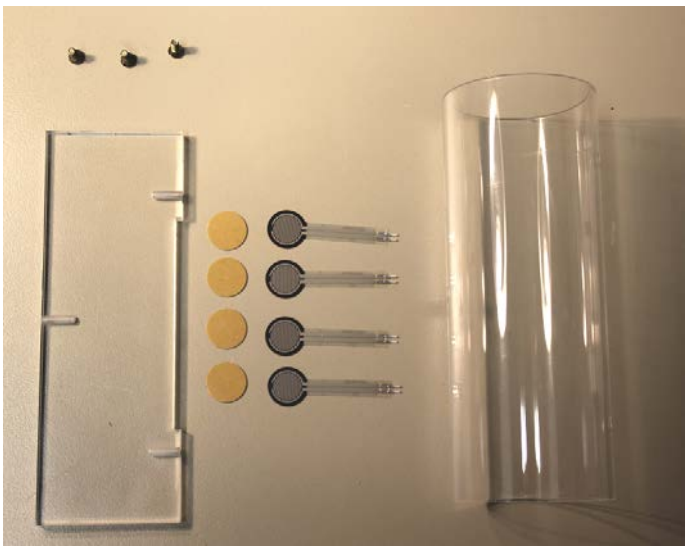

Abbildung 4: Acrylplatte links, Sensoren mittig und Acrylrohr rechts

Auf der einen Seite der Zwischenebene befindet sich die Platine mit den elektronischen Komponenten und auf der anderen Seite die Drucksensoren. Der Druck der Ventiltasten wird in das Innere des Rohres und über kompressible Silikonscheiben auf die Drucksensoren übertragen. Durch das weiche Material ergibt sich für die Ventiltasten ein Weg von maximal 5 mm. Die Zwischenebene hat auf der einen Seite eine Aussparung, um die Sensoren mit der Platine zu verbinden. Die Drucksensoren werden zur Fixierung zwischen zwei Holzplatten eingespannt, wobei die obere, runde Aussparungen mit einem Durchmesser 18 mm für deren aktive Fläche hat. Zwischen den Platten wurde die Flächen der Sensoren um etwa 0.3 mm ausgespart. Dadurch sind die Sensoren fixiert und müssen nicht angeklebt werden.

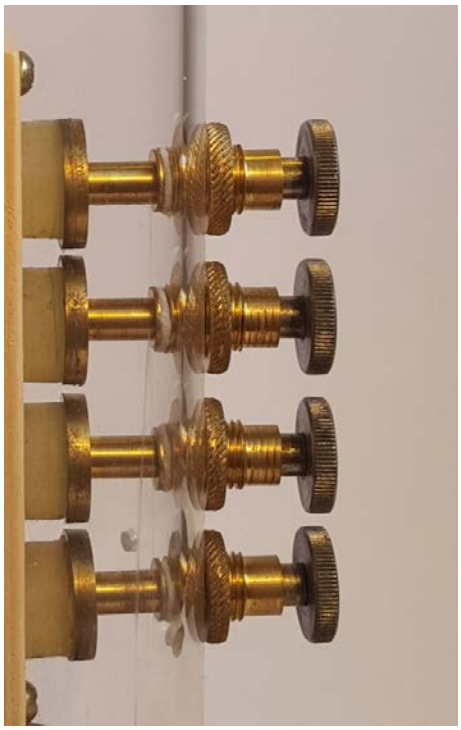

Abbildung 5: Ventiltasten, welche mit dem Acrylrohr verschraubt sind

Die Höhe der Bolzen der Ventiltasten ist über ein Gewinde einstellbar und so justiert, dass die Silikonscheiben nicht herausfallen können und die Drucksensoren noch nicht belastet werden. Dadurch sind die Scheiben mechanisch fixiert und die Tasten lassen sich noch drehen. Durch Schüttelbewegungen können diese jedoch ungewollt Druck auf die Sensoren ausüben. Dies muss bei der Auswertung der Sensordaten berücksichtigt werden.
Auf der Rückseite der Zwischenebene ist die Elektronik untergebracht, diese besteht im Wesentlichen aus dem Teensy Board und einer weiteren Platine mit der notwendigen Beschaltung und wird in dem Kapitel 3.1.4 im Detail beschrieben.
Die Piezoschalter für die Oktavauswahl werden zur Befestigung in einer Messingplatte untergebracht. Diese hat drei Löcher, mit einem Durchmesser von 16 mm. Die Schalter haben ein rundes Aluminiumgehäuse mit Gewinde. Darüber werden sie mit einer Mutter auf der Rückseite der Messingplatte verschraubt. Zur Unterbringung des Moduls wird die entsprechende Fläche der Rohrwand abgeflacht und teilweise ausgespart. Hier wird die Messingplatte mit den Schaltern eingesetzt und mit dem Rohr verschraubt. In der Abbildung 6 sind die Oktavschalter auf der Rückseite des Controllers zu sehen.

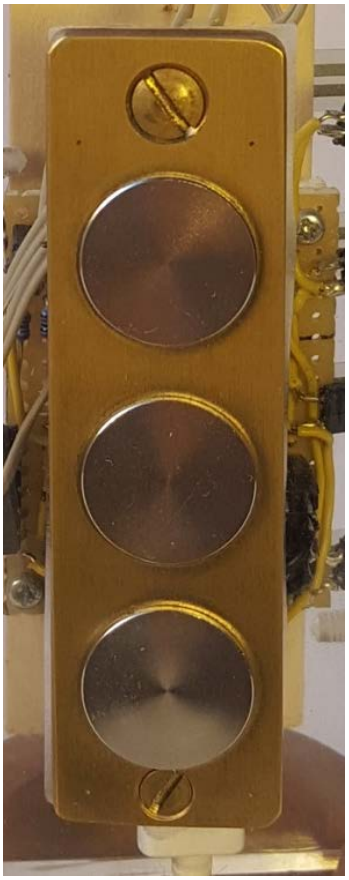

Abbildung 6: Piezoschalter mit der Messingplatte

Am oberen und unteren Ende ist das Rohr durch Holz abgeschlossen. Der Umfang dieser Teile ist so gestaltet, dass sie durch Stecken mit dem Rohr verbunden werden können und dadurch fest sitzen. Allerdings kann sich das Volumen des Holzes durch den Einfluss von Luftfeuchtigkeit ändern. Der Durchmesser des oberen Rundholzes verringert sich an einer Stelle, um das Instrument komfortabel halten zu können. Im Inneren ist eine Bohrung für die Kabelverbindungen der oberen Drucksensoren mit der Platine. Durch die Bohrung wird auch die obere Holzscheibe, die in dem Kopfteil des Controllers sitzt, durch eine Gummischnur fixiert.
In dem oberen Holzzylinder befindet sich das Excitation Pad. Hier schließt eine frei bewegliche Holzscheibe, den Controller nach oben hin ab. Unter dieser sitzen vier Silikonscheiben, welche den Druck auf die darunter liegenden Drucksensoren übertragen. Diese sind im Abstand von jeweils 90°, am Rand des Hohlraumes angeordnet (siehe Abbildung 7). Bei geradem Herunterdrücken der Holzscheibe werden alle vier Sensoren gleich stark belastet. Bei leicht schrägem Druck, wird die Belastung je nach Richtung unterschiedlich auf die Sensoren verteilt. Um zu vermeiden, dass die Silikonpuffer verrutschen, sind sie in eine flache runde Silikonfläche eingelassen, welche die Fläche unter der Holzscheibe abdeckt.

Abbildung 7: Silikonpuffer und Sensoren zur Steuerung der Anregung

Im Anhang B ist in Abbildung 20 der schematische Querschnitt des Instrumentes dargestellt. Aus den unterschiedlichen Farben gehen die verwendeten Materialien Silikon, Acryl, Holz, Metall und Leiterplatte (PCB) hervor.

3.1.4. Elektronische Umsetzung

Als Basis für die elektronische Umsetzung wird das USB-Entwicklungssystem Teensy 3.1 verwendet (Stoffregen und Coon, 2015). Auf der Platine mit den Maßen 35 mm x 18 mm ist der 32 Bit ARM Cortex-M4 Mikroprozessor MK20DX256, mit der notwendigen Beschaltung und ein USB-Anschluss untergebracht. Die Anschlüsse des Prozessors sind über Lötpunkte einfach zugänglich, sodass die Schaltung um zusätzliche Komponenten erweitert werden kann. Über die Teensyduino Erweiterung der Arduino Entwicklungsumgebung kann der Programm-Code für das Teensy Board kompiliert werden. Das übertragen der Firmware erfolgt direkt über die USB-Schnittstelle.

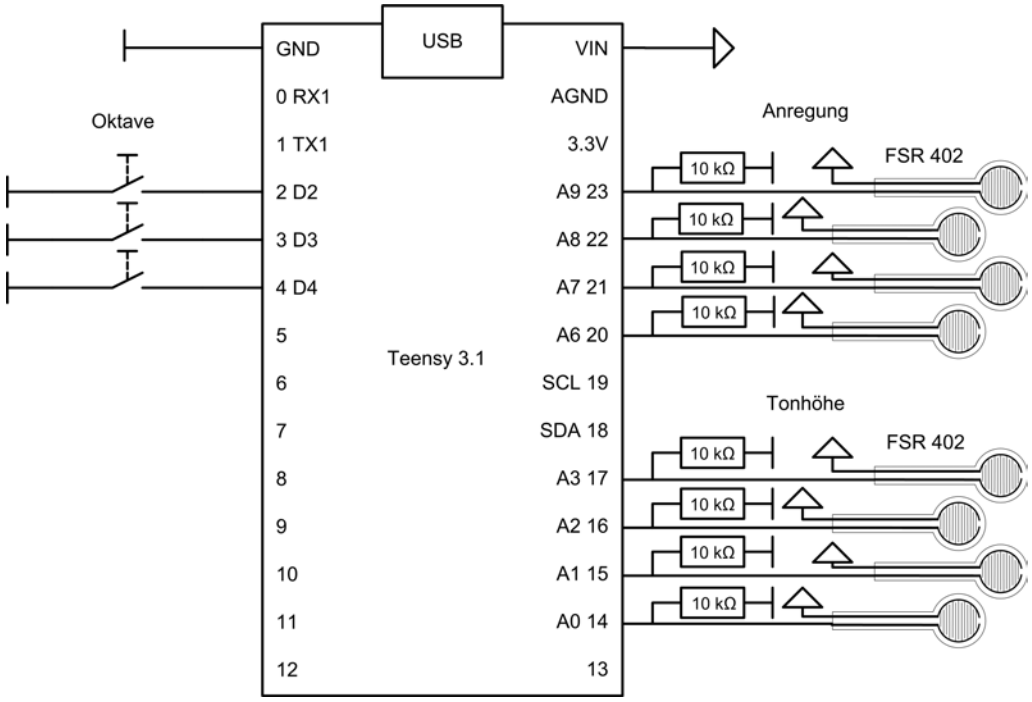

Abbildung 8: Teensy-Board erweitert um die Sensoren und notwendigen Bauteile

In Abbildung 8 ist das Teensy-Board mit der verwendeten Beschaltung dargestellt. Diese besteht aus vier Drucksensoren (FRS 402) zur Tonhöhenauswahl und vier zur Steuerung der Anregung. Sie bilden mit je einem 10 kΩ Widerstand einen Spannungsteiler. Dieser teilt die Spannung an dem jeweiligen Analog-Digital-Converter (ADC) durch dem variablen druckempfindlichen Widerstand (FSR) und den festen 10 kΩ Widerstand, zwischen der 5 V Spannungsversorgung und der Masse. Damit ändert sich die am ADC anliegende Spannung im Verhältnis zu dem Druck des FSR, entsprechend der Formel 1 in Kapitel 2.3.2.
Für die Anschlüsse der Piezoschalter zur Oktavauswahl werden die internen Pullup-Widerstände des Mikroprozessors aktiviert. Dadurch können die Schalter direkt mit dem Anschluss und Masse verbunden werden. Für die steckbaren Verbindungen zwischen der Platine, den Sensoren und Schaltern sind an deren Kabelenden Stiftleisten und an der Platine abgewinkelte Federleisten angelötet.
Die Spannungsversorgung und Datenübertragung zu dem PC erfolgt über den Micro-USB-Anschluss. An dem PC wird das Teensy-Board nach entsprechender Programmierung als MIDI-Gerät erkannt.

3.1.5. Firmware

Die Firmware besteht im wesentlichen aus einer initialen Konfiguration des Mikroprozessors und der Initialisierung der globalen Variablen. Anschließend wird in dem Programm dauerhaft eine Schleife durchlaufen, bis die Reset-Taste gedrückt oder die Stromversorgung unterbrochen wird. Innerhalb der Schleife wird erst die Funktion zum Einlesen der Tonhöhe aufgerufen. In den folgenden Schleifendurchläufen wird durch einen Zähler ein Zeitfenster abgewartet. Innerhalb dieser Zeit wird überprüft, ob sich die zuvor ausgewählte Note ändert, falls nicht alle Tasten simultan gedrückt wurden. Bevor die Note übertragen wird, erfolgt ein Aufruf der Funktion zur Abfrage der Oktavschalter und es werden die an den ADCs anliegenden Spannungen der Drucksensoren zum Auslösen der Note und zur Modulation des Klanges eingelesen. Anschließend wird die Ton-

höhe ermittelt und je nach Status die entsprechenden MIDI-Befehle über die USB-Schnittstelle übertragen.
Wenn momentan keine Note gespielt wird, erfolgt ein Vergleich des Mittelwertes der vier Sensoren zum Auslösen der Note mit einem Schwellenwert. Ist dieser überschritten, wird ein Note-On Befehl gesendet. Wird momentan eine Note gespielt, wird zunächst überprüft ob der Wert des mittleren Druckes einen Schwellenwert zum Beenden der Note unterschreitet. In diesem Fall wird die aktuelle Note durch ein Note-Off Befehl beendet. Wird der Schwellenwert nicht unterschritten, wird überprüft, ob sich die Tonhöhe geändert hat oder die gleiche Note noch einmal ausgewählt wurde. Ist dies der Fall wird der Note-On Befehl mit der aktuellen Tonhöhe übertragen und die vorherige Note beendet.

3.1.5.1. Tonhöhensteuerung

Die Spannungen, welche im Verhältnis zu dem anliegenden Druck der vier Ventiltasten stehen (siehe 2.3.2) werden in jedem Schleifendurchlauf über die 12-Bit ADCs des Mikroprozessors abgetastet. Übersteigt der digitale Wert einen festen Schwellenwert, wird diese Taste als gedrückt erkannt und das entsprechende Bit des Notenwertes gesetzt. Ein recht hoher Schwellenwert `NOTETHR` von 200, was etwa einer Spannung von 0.98 V entspricht, soll vermeiden, dass die Taste ungewollt als gedrückt erkannt wird. Dies reduziert die Dynamik, jedoch unterstützt das MIDI-Protokoll ohnehin nicht die volle Dynamik der ADCs. Die MIDI-Befehle stellen nur eine Wortbreite von 7 Bit für die Datenübertragung zur Verfügung, da das erste Bit der 8-Bit Befehle zwischen Status- und Datenbytes unterscheidet.
Die Argumente beim Aufruf der Funktion zum Einlesen der Tonhöhe sind die Zeiger auf den Wert der Note und des Tastendruckes. Der Wert der Note wird vor dem Funktionsaufruf auf Null gesetzt, d.h. alle Bits sind gelöscht. Die Funktion liest in einer Schleife die digitalen Werte der Spannungen der vier FSR-Sensoren ein. Wenn einer den Schwellenwert überschritten hat, wird das entsprechende Bit des Notenwertes, durch die Arduino-Funktion `bitWrite()`, gesetzt. Der mittlere Tastendruck errechnet sich aus der Summe aller Werte, welche den Schwellenwert überschreiten, im Verhältnis zu der Anzahl der gedrückten Tasten. Zur Übertragung des mittleren Tastendruckes, als MIDI-Controller Befehl, wird der Wert skaliert und auf den Wertebereich von 0 bis 127 begrenzt. Dem folgenden Quellcode-Ausschnitt 1 ist die Funktion `getNote()`, zum Einlesen der Tonhöhe, zu entnehmen.

```
 1: /* getNote: Reading analog values set binary note and key pressure
 2:    Input parameters: Pointer to binaryNote
 3:                      Pointer to meanKeyPressure
 4:  */
 5: void getNote(int* binaryNote, int* meanKeyPressure)
 6: {
 7:   // Number of selected keys
 8:   int numKey = 0;
 9:   // Loop over all keys
10:   for(int k = 0; k<4; k++)
11:   {
12:      // Read analog values
13:      analogKeyFSR[k] = analogRead(analogKeyPins[k]);
14:      // Set Bites according to pressed keys
15:     if(analogKeyFSR[k] > NOTETHR)
16:     {
17:       bitWrite(*binaryNote, 3-k, 1);
```

```
      *meanKeyPressure += analogKeyFSR[k]-NOTETHR;
      numKey++;
    }
  }
  // Average pressure, dividing sum by number of pressed keys if any pressed
  if (numKey > 0)
    *meanKeyPressure = *meanKeyPressure/numKey;
  else
    *meanKeyPressure = 0;
  // Map to MIDI range
  *meanKeyPressure = map(*meanKeyPressure, 0, 1023-NOTETHR, 0, 127);
  // Limit to 7 LSB
  *meanKeyPressure &= 0x007F;
}
```

Quellcode 1: Arduino Quellcode der Funktion zum Einlesen der Tonhöhe

Um den Tonhöhenumfang auf drei Oktaven zu erweitern, wird in jedem Schleifendurchlauf der Status der drei Piezoschalter abgefragt. Durch die Verwendung der internen Pull-Up Widerstände ist der Status des Anschlusses im nicht gedrückten Zustand `HIGH`. Wenn ein Schalter gedrückt ist, wechselt der Zustand des Anschlusses auf `LOW` und die entsprechende Oktave wird selektiert. D.h. die ausgewählte Tonhöhe bleibt unverändert oder wird um 12 oder 24 Halbtonschritte erhöht. Es wird stets die höchste Oktave ausgewählt, wenn mehrere Schalter gedrückt sind. Die Funktion zur Steuerung der Oktave ist in dem Quellcode-Ausschnitt 2 zu sehen.

```
/* Octave */
void getOctave(int* octave)
{
  // Loop over all keys
  for(int n = 0; n < numButtons; n++){
    // Set octave to button number which is low
    if (digitalRead(digitalPins[n]) == LOW)
      *octave = n;
  }
}
```

Quellcode 2: Funktion zur Steuerung der Oktave

Die ausgewählte Oktave ändert sich erst wenn ein anderer Schalter gedrückt wurde. Um die MIDI-Notennummer der ausgewählten Tonhöhe zu bestimmen, wird die Nummer der Oktave mit zwölf multipliziert und mit der tiefsten Note (`BASENOTE`), mit einem festen Wert von 36, sowie der binär ausgewählten Tonhöhe summiert. Da die binär ausgewählte Note bei eins beginnt, muss von der Summe noch eins abgezogen werden um die momentane Tonhöhe zu erhalten:

```
  int note = binaryNoteSend + 12*octave + BASENOTE - 1;
```

Quellcode 3: Berechnung der momentanen Tonhöhe

Um beispielsweise den tiefsten Ton C3 zu spielen, ist der Zeigefinger gedrückt, was der Binärzahl `Bx0001` entspricht. Wenn die erste Oktavtaste ausgewählt ist, wird die zwölf mit null multipliziert. Wird von der Summe eins subtrahiert ergibt sich die Notennummer des tiefsten Tones, 36 (siehe Tabelle 4).
Die Schwierigkeit der Tonhöhenauswahl liegt darin, alle benötigten Ventiltasten möglichst synchron zu betätigen. Wenn das nicht gelingt, wird für den Übergang, bis alle Tasten gedrückt sind, ungewollt eine falsche Tonhöhe ausgewählt. Ein Zeitfenster von 150 Schleifendurchläufen soll diese Zwischentöne verhindern. Innerhalb des Zeitfensters müssen die beteiligten Tasten gedrückt

werden. Da die Dauer eines Schleifendurchlaufs von den Verzweigungen des Programmcodes abhängt, wird die tatsächliche Länge des Zeitfensters experimentell ermittelt. Diese wird bei der Ausführung des Programmcodes während 60 Wiederholungen der Tonhöhenauswahl ermittelt und ausgegeben. Der Programmcode wird dazu entsprechend angepasst. Das Zeitfenster hat im Mittel eine Dauer von 46 ms (SD = 2.56).

3.1.5.2. Anregung und Modulation

Zusätzlich zu der Tonhöhensteuerung ist es notwendig, die Note auszulösen und dabei die Intensität des Klanges zu steuern. Dazu dient das Excitation Pad am oberen Ende des Controllers. Der Druck wird über Silikonpuffer auf die vier darunter liegende Drucksensoren übertragen. Die eingelesenen Werte werden gemittelt und als MIDI-Anschlagstärke übertragen und zur kontinuierlichen Steuerung der Amplitude verwendet. Die Sensordaten werden auch verwendet um weitere Klangparameter interaktiv steuern zu können. Dazu wird ermittelt wie unterschiedlich der Druck auf die Sensoren verteilt ist.
Die relativen Differenzen des Drucks der gegenüberliegenden Sensoren ergeben sich aus dem Verhältnis der Differenz zu der Summe der jeweiligen Werte. Dadurch ergeben sich zwei Richtungsinformationen, welche unabhängig von dem gesamten Druck sind. Für die Berechnung wird folgende Formel verwendet:

$$dirCtrl = 63 \cdot \frac{contrValue[0] - contrValue[2]}{contrValue[0] + contrValue[2]} + 63 \tag{3}$$

Die Werte werden so skaliert, dass sie zwischen 0 und 127 liegen. Da die relativen Differenzen durch leicht schräges Drücken stark schwanken können, werden diese durch einen rekursiven Tiefpassfilter erster Ordnung geglättet:

$$y[n] = (1 - \alpha) \cdot x[n] + \alpha \cdot y[n - 1]; \tag{4}$$

Für die Filterung findet eine explizite Typumwandlung der Werte von Integer zu Float statt. Experimente haben gezeigt, dass die Steuerung über relative Differenzen sehr sensibel ist. Es zeigt sich, dass durch die Implementierung der absoluten Differenz der gegenüberliegenden Sensoren der Anregung die Parameter der Klangsynthese genauer gesteuert werden können:

```
1:  int xdir = map(ctrlVal[2] - ctrlVal[0], -1023, 1023, 0, 127);
```

Quellcode 4: Berechnung der alsoluten Differenzen

Eine Verbesserung bei der Bestimmung der Anschlagstärke wird erzielt, wenn neben dem überschreiten eines Schwellenwertes des durchschnittlichen Druckes zur Anregung, die Werte abgetastet werden, bis das erste Maximum erreicht ist. Die Steuerdaten zur Modulation des Klanges werden nur alle 200 Schleifendurchläufe und im Falle einer Änderung übertragen. Dies erweist sich als ausreichend für die Steuerung und reduziert den Datentransfer. Zusätzlich wird zur Steuerung der Klangsynthese der mittlere Tastendruck der Ventiltasten als MIDI-Controller Daten an den Computer gesendet.

3.1.5.3. Datenübertragung an den PC

Zur Übertragung der Steuerinformationen an den Computer wird die MIDI-Schnittstelle verwendet. Das USB-Entwicklungssystem Teensy 3.1 bietet den Vorteil, das es von dem angeschlossenen Computer direkt als MIDI-Gerät erkannt wird. In der Teensyduino Erweiterung der Arduino IDE

stehen Funktionen bereit um Daten über die USB MIDI-Schnittstelle zu übertragen (Stoffregen und Coon, 2015). In dem folgenden Quellcode-Auszügen werden Beispiele der Funktionen dargestellt:

```
1: // Send MIDI controller data
2: usbMIDI.sendControlChange(controller[0], dirCtrl, channel);
3: // Send MIDI note on command
4: usbMIDI.sendNoteOn(note, velocity, channel);
5: // Send MIDI note off command
6: usbMIDI.sendNoteOff(note, 0, channel);
```

Quellcode 5: Beispiele von Funktionen zur Übertragung der MIDI-Befehle über die USB-Schnittstelle

Durch diese Funktionen können Befehle für Note an/aus oder Controller-Daten über die USB-MIDI-Schnittstelle übertragen werden. Alternativ zum MIDI-Protokoll, ist es möglich die Daten über eine andere serielle Schnittstelle zu übertragen. Allerdings müsste dazu die Implementierung angepasst werden.

3.1.6. Klangsynthese

Die Klangsynthese wird mit Pure Data (Puckette et al., 1996) realisiert und besteht aus einem Patch zum Empfangen der MIDI-Daten, zur Steuerung der Klangsynthese und zur Klangverarbeitung. Die Synthese ist in zwei Subpatches unterteilt. Das eine generiert Sägezahnfunktionen aus 24 Harmonischen. Das übergeordnete Subpatch steuert zwölf dieser Generatoren parallel an und gibt die gemittelten Signale aus. Die Klangsynthese verfügt für den linken und rechten Kanal jeweils über zwölf Sägezahn-Generatoren. Die Grundfrequenzen der Generatoren werden durch die ausgewählte Tonhöhe gesteuert und jeweils durch ein Zufallssignal leicht moduliert. Dadurch entsteht für jeden Kanal ein unabhängiger Unisono-Effekt, dessen Intensität durch die Amplitude der Zufallssignale steuerbar ist.
Eine ADSR-Hüllkurve steuert die Amplitude des linken und rechten Kanals. Da eine interaktive Steuerung der Amplitude möglich sein soll, wird die Hüllkurve mit den Daten des MIDI-Controller gewichtet. Die Signale beider Kanäle durchlaufen anschließend jeweils einen Moog-Tiefpassfilter. Dessen Grenzfrequenz ebenfalls durch eine Hüllkurve gesteuert wird, welche interaktiv gewichtet ist. Zusätzlich ist die Grenzfrequenz durch einen LFO modulierbar. Der Q-Faktor des Filters ist ebenfalls variabel.

3.1.7. Mapping

Zur Steuerung der Tonhöhe wird die MIDI-Notennummer in die entsprechende Frequenz in Hertz umgerechnet. Diese bestimmt die Grundfrequenz der Klangsynthese. Ein Note-On Befehl Triggert die Hüllkurve der Amplitude und der Grenzfrequenz des Tiefpassfilters. Die Gewichtung der Amplituden-Hüllkurve wird über den mittleren Druck auf die obere Holzfläche gesteuert. Die Schräge des Druckes steuert in der einen Richtung die Gewichtung der Filter-Hüllkurve und in der anderen den Q-Faktor des Tiefpassfilters. Die Daten des Controllers werden zuvor geglättet und skaliert. Bei der Skalierung wird der minimale und maximale Wert der Daten auf einen vorgegebenen Wertebereich abgebildet. Daten zur Steuerung der Grenzfrequenz und der Q-Faktor werden als vertikale und horizontale Linien in einem Diagramm dargestellt. Dadurch lässt sich die Schräge des Drucks auch visuell überprüfen.
Über den Tastendruck der Ventiltasten wird die Intensität des Unisono-Effekts gesteuert. Durch ei-

ne exponentielle Funktion wird der Effekt erst bei sehr festem Tastendruck deutlich wahrnehmbar. Dann wird die Grenzfrequenz des Filters zusätzlich durch einen LFO moduliert.

3.1.8. Spielweise

Die Steuerung der Tonhöhe erfolgt durch die Tastenkombinationen der vier Finger der bevorzugten Hand. Um die Oktave zu wechseln muss mit dem Daumen der gleichen Hand ein Piezo-Schalter auf der gegenüberliegenden Seite betätigt werden. Nur wenn eine oder mehrere Ventiltasten gedrückt sind, kann durch Drücken auf die obere Scheibe die entsprechende Note ausgelöst werden. Während der Ton gespielt wird, kann über den Aftertouch der Ventiltasten der Unisono-Effekt gesteuert werden.
Wird der Druck auf die obere Scheibe senkrecht ausgeübt, befinden sich die Grenzfrequenz und der Q-Faktor des Tiefpassfilters im mittleren Bereich. Durch Verlagern des Druckes in eine beliebige Richtung können die Filterparameter beeinflusst werden. Wenn der Druck auf die obere Fläche aufrecht gehalten wird, können die Ventiltasten losgelassen werden und der Ton hält weiterhin an. Das Auswählen einer neuen Tastenkombination verändert die Tonhöhe des gespielten Klanges. Sobald der Druck auf die obere Fläche aufgehoben wird, endet der Ton. Der Druck steuert die Amplitude, wodurch zusätzlich zu der vorgegeben Hüllkurve die Klangintensität reguliert werden kann.

3.2. Evaluationsmethode

Die Besonderheit des entwickelten Controllers ist die Steuerung der Tonhöhe über Vier-Finger-Kombinationen. Diese soll bei der Evaluation im Vordergrund stehen.
Im Vergleich zu der Tonhöhensteuerung bei einer Gitarre, einem Cello oder Klavier ist es nicht notwendig die Position der Hand zu verändern. Sie bleibt stets an der gleichen Stelle und die Finger sind fest den vier Tasten zugewiesen. Das Anheben oder Absenken einzelner oder mehrerer Finger ist ausreichend um eine Tonhöhe auszuwählen. Dadurch entfällt die Bewegungszeit, um eine entsprechende Position zu erreichen. Sofern möglich optimieren die Musiker ihre Spielweise, um möglichst geringe Wege bei einem Tonwechsel zurückzulegen.
Da bei der Tonhöhensteuerung über Vier-Finger-Kombinationen keine Bewegungszeit notwendig ist, wird vermutet, dass diese Art der Tonhöhensteuerung effizienter ist. Jedoch ist es erforderlich, dass mehrere Finger zur gleichen Zeit bewegt werden um die entsprechenden Tasten möglichst simultan zu drücken. Dazu muss das motorische System auf die Bewegungen vorbereitet, bzw. „programmiert“ werden. Die Dauer für dieses Programmieren der Bewegung, hängt von deren Komplexität ab (Karnath und Thier, 2006; S. 290). Thon und Bonneviale (1995) untersuchen die Reaktionszeiten von Einzelfingerantworten und Zwei-Finger-Kombinationen. Die Ergebnisse zeigen, dass bestimmte Fingerkombinationen, wie beispielsweise kleiner Finger und Mittelfinger zu deutlich höheren Reaktionszeiten führen. Die durchschnittliche Reaktionszeit für Antworten mit zwei Fingern einer Hand beträgt 1076 ms.
Bei der Klaviatur muss zur Tonhöhensteuerung stets die richtige Taste ausgewählt werden. Nach dem Hickschen Gesetz steigt die Reaktionszeit bei der eine Auswahl getroffen werden muss, mit zunehmender Anzahl an Optionen logarithmisch an Hick (1952). Bei den Experimenten ruhen die zehn Finger beider Hände bereits auf den Tasten, um die Antwort einzugeben. Die Reaktionszeit steigt von etwa 180 ms, wenn keine Auswahl getroffen wird, auf etwa 620 ms bei zehn möglichen Alternativen (Hommel und Nattkemper, 2011; S. 103). Bei der Tonhöhenauswahl mit der Klaviatur ruhen die Finger meist nicht auf den Tasten, sondern bewegen sich von einer zur nächsten. Die benötigte Zeit für eine gezielte Bewegung hängt von der Entfernung und Größe der Zielfläche ab

(MacKenzie, 1992). Bei der Klaviatur beträgt die maximale Distanz innerhalb einer Oktave etwa 140 mm. Die Fläche beträgt für weiße Tasten etwa 45 mm x 23.6 mm und für die schwarzen etwa 100 mm x 12.7 mm.
Dies zeigt, dass die Dauer um eine Tonhöhe auszuwählen nicht leicht vorhergesagt werden kann. Um die Effizienz der Tonhöhensteuerung über Vier-Finger-Kombinationen bewerten zu können ist es erforderlich ein Experiment durchzuführen und dabei unterschiedliche Arten der Tonhöhensteuerung zu vergleichen. Die Klaviatur ist die am häufigsten verwendete Benutzerschnittstelle für Anwendungen der Computer-Musik (Holland et al., 2013b; S. 123) und wird auch häufig als Melodieinstrument eingesetzt. Aus diesem Grund wird diese für den Vergleich verwendet.
Die beiden Arten der Tonhöhenauswahl weisen starke Unterschiede auf. Bei der Klaviatur gibt es keine feste Zuweisung zwischen Fingern und Tasten. Die Hand bewegt sich zu der Position der Taste. Um einen Ton zu spielen, wird nur eine Taste gedrückt. Außer bei den Tonhöhen C, C#, D# und G müssen hingegen beim Controller mehrere Tasten gleichzeitig gedrückt werden. Außerdem wird bei der Klaviatur der Ton durch das Drücken der Taste gespielt. Bei dem Controller wird dadurch ausschließlich die Tonhöhe ausgewählt. Um den Ton zu spielen, muss zusätzlich der Auslöser, am oberen Ende des Controllers gedrückt werden.
Diese Unterschiede erschweren den direkten Vergleich. Zudem ist es unklar, welcher Unterschied die Spielgeschwindigkeit und Fehlerrate stärker beeinflusst. Um dies zu berücksichtigten, wird der Controller in zwei unterschiedlichen Varianten mit der Klaviatur verglichen. Bei der ersten Variante, wird der Ton direkt durch den Tastendruck und bei der zweiten durch zusätzliches Drücken auf die obere Fläche gespielt.
Es soll bei dem Experiment die Hypothese untersucht werden, ob über Vier-Finger-Kombinationen die Tonhöhen schneller ausgewählt werden können als mit der Klaviatur. Dazu werden die mittleren Reaktionszeiten der beide Varianten des Controllers mit denen der Klaviatur verglichen und mittels der einfaktoriellen Varianzanalyse für abhängige Stichproben auf signifikante Unterschiede überprüft. In dem Reaktionstest werden den Probanden unterschiedliche Tonhöhen als visuelle Stimuli präsentiert. Der entsprechende Ton soll daraufhin so schnell und fehlerfrei wie möglich gespielt werden.
Um den Unterschied der festen Tastenzuweisung und der Fingerkombinationen zu berücksichtigen, wird der Versuch für eine zusätzlich Nebenbedingung durchgeführt. Bei dieser ist die Spielweise für beide Interfaces identisch. Dies wird ermöglicht, indem der Tonumfang auf vier Töne beschränkt wird, da nur vier Finger einer Hand zur Tonhöhenauswahl zur Verfügung stehen. Die Finger werden auch bei der Klaviatur fest vier Tasten zugewiesen. Beim Controller werden keine Fingerkombinationen zugelassen. Damit wird eine Versuchsbedingung geschaffen, bei der die Ergebnisse zwischen den beiden Interfaces verglichen werden können.
Auch die Reaktionszeiten für die Nebenbedingung mit vier Noten werden zwischen der Klaviatur und den beiden Varianten des Controllers verglichen und auf signifikante Unterschiede überprüft. Für diese Bedingung ist die Spielweise für Klaviatur und für die Variante des Controllers, bei der die Note direkt über den Tastendruck gespielt wird, identisch. Die einzigen Unterschiede liegen in der Mechanik der Taste und der Position der Hand. Diese ist für die Klaviatur horizontal und für den Controller vertikal. Da vermutet wird, dass die Klaviatur für die Mehrheit der Versuchsteilnehmer das vertrautere Interface darstellt, wird die Hypothese untersucht, ob die Reaktionszeiten für die Klaviatur geringer ist, als für den Controller. Beim Vergleich der Reaktionszeiten beider Varianten des Controllers wird erwartet, dass für die Variante, bei der zusätzlich Druck auf die obere Fläche ausgeübt werden muss, um den Ton zu spielen, signifikant höhere Werte gemessen werden.
Damit sich die Versuchspersonen mit dem jeweiligen Interface und dem Versuchsablauf vertraut

machen können, ist eine kurze Trainingsphase vor dem jeweiligen Versuchsdurchlauf vorgesehen. Der Reaktionstest wird für eine geringere Anzahl an Wiederholungen vor dem eigentlichen Versuch als Trainingsphase durchgeführt. Bei dem Vergleich der unterschiedlichen Tonhöhen innerhalb eines Interfaces wird für die Klaviatur kein Unterschied zwischen den zwölf oder vier Tonhöhen erwartet. Jedoch sind bei dem Controller manche Fingerkombinationen schwerer zu greifen (Thon und Bonneviale, 1995). Für diese sind deutlich höhere Reaktionszeiten zu erwarten. Der Speed-Accuracy Tradeoff beschreibt die Abhängigkeit zwischen der Geschwindigkeit und der Genauigkeit mit der eine Aufgabe erfüllt wird (Pew, 1969). Demnach führt eine schnellere Ausführung einer Bewegung zu mehr Fehlern. Durch diesen Zusammenhang müssen neben der Reaktionszeit auch die Fehler betrachtet werden, welche beim Reaktionstest auftreten. Es kann ein oder mehrere Male die falsche Note gespielt werden, bevor die vorgegebene Note gespielt wird. Bei der Tonhöhenauswahl über Fingerkombinationen sind die zu drückenden Tasten zeitgleich zu betätigen, ansonsten werden ungewollte Zwischentöne gespielt. Dies ist der Grund, warum für den Reaktionstest mit Vier-Finger-Kombinationen mehr Fehler auftreten können als für die Klaviatur. Insbesondere für die zweite Variante werden mehr Fehler erwartet. Für die Nebenbedingung mit nur vier Tonhöhen ist kein Unterschied zu erwarten.
Die unabhängigen Variablen sind die Interface-Variante und die Anzahl der zu spielenden Noten. Die Variable Interface-Variante hat die Ausprägungen Klaviatur, Ventiltasten mit direktem Spielen des Tons und zusätzlichen Drücken auf die obere Fläche. Die Anzahl der zu spielenden Noten sind vier oder zwölf, alle Halbtöne einer Oktave. Die abhängigen Variablen sind die Reaktionszeit, um die richtige Tonhöhe zu spielen und die Anzahl der Fehler. Als Fehler wird gewertet, wenn nicht gleich die richtige Tonhöhe gespielt wird, sondern erst ein oder mehrere falsche. Außerdem soll der subjektive Eindruck der Schwierigkeit, Zuverlässigkeit und der benötigten motorischen Fähigkeiten des Reaktionstests erfasst werden, um diese mit den Reaktionszeiten und Fehlerraten zu vergleichen. Hier ist zu erwarten, dass die subjektiven Daten signifikant mit den gemessenen korrelieren.
Eine Aussage über das Instrument als Ganzes kann durch den Reaktionstest alleine nicht getroffen werden. Dazu ist es notwendig, dass der Nutzer sich damit vertraut macht und den vollen Funktionsumfang kennenlernt. Neben der Tonhöhensteuerung ist die Modulation des Klanges ein wichtiger Aspekt des digitalen Musikinstrumentes. Die Versuchsteilnehmer sollen auch die Möglichkeit bekommen, nach belieben mit dem Instrument zu spielen. Es wird darauf verzichtet, konkrete Aufgaben zu stellen. Die Probanden können sich dadurch auf den kreativen Prozess konzentrieren und die Ergebnisse der anschließenden Bewertung sind weniger von der musikalischen Erfahrung abhängig. In einem Fragebogen sollen einzelne Aspekte des Instrumentes und die Erfahrungen beim freien Spielen bewertet werden. Am Ende des Fragebogens haben die Versuchsteilnehmer die Möglichkeit, ihr Feedback zu dem Instrument und Verbesserungsvorschläge zu formulieren.
Ein weiterer Fragebogen soll dazu dienen, die Usability des Instrumentes einzuschätzen und Probleme bei der Bedienung aufzudecken. Dazu werden die Fragen der System Usability Scale so angepasst, dass sie sich auf das Instrument beziehen und die Bedeutung erhalten bleibt. Abschließend werden die Teilnehmer zu ihrer musikalischen Erfahrung befragt. Dazu werden die Fragen des Goldsmiths Musical Sophistication Index (Gold-MSI) verwendet, welche sich auf das musikalische Training beziehen (Müllensiefen et al., 2012). Diese sollen Aufschluss über das erlernte Instrument, die musikalische Ausbildung und die Intensität des Trainings geben.
Die Untersuchungsmethode besteht aus einem Versuch, um die Tonhöhensteuerung zu bewerten und einem kreativen Teil, indem die Möglichkeiten des Instrumentes erlebt werden können.

3.2.1. Versuchsaufbau

Die Versuche fanden im MediaLab des Fachgebiets Audiokommunikation an der Technischen Universität Berlin statt. Die Versuchspersonen saß an einem Tisch, auf dem sich ein MIDI-Keyboard (Evolution MK-125), der im Rahmen der Masterarbeit realisierte Controller, ein Computermonitor und zwei Aktivlautsprecher (Fostex 6301B) befanden. Diese Geräte waren mit dem Computer des Versuchsleiters verbunden und der Monitor wurde als Zweitbildschirm angesteuert. Für die Versuche wurde ein Notebook (IBM X220) mit 2.5 GHz Prozessor und 8 GB RAM verwendet. Auf dem 64-Bit System ist das Betriebssystem Windows 7 und unter anderem Pure Data (Puckette et al., 1996) installiert. Als Soundkarte diente das RME Babyface Audio-Interface mit MIDI-Schnittstelle, welche über ASIO-Treiber angesteuert wurde. Das MIDI-Keyboard war mit dem MIDI-Eingang des Audio-Interfaces und die Aktivlautsprecher über den Audioausgang verbunden. Der Versuchsaufbau ist in Abbildung 9 dargestellt.

Abbildung 9: Versuchsaufbau

Bei dem Controller wurden die MIDI-Daten über die USB-Schnittstelle des Teensy-Boards und beim Keyboard über die MIDI-Schnittstelle der Soundkarte an den PC übertragen. Da der Controller in zwei unterschiedlichen Varianten verwendet wurde, musste zwischen den Reaktionstests ein Aktualisierung der Firmware des Controllers über die Arduino IDE erfolgen. Anschließend konnte der Controller wieder als USB-MIDI-Gerät genutzte werden.
Die Grafikerweiterung Graphics Environment for Multimedia (GEM), welche in Pure Data integriert ist (Danks, 1997), erzeugte die visuellen Stimuli. Die Reihenfolge der Tonhöhen sind zufällige Permutationen des Tonumfangs und wurden zuvor mit Matlab generiert und während der Versuche von Pure Data aus einer Textdatei eingelesen. Die eintreffenden MIDI-Noten wurden über eine digitale Klangsynthese in ein akustisches Feedback gewandelt. Dieses wurde über die Soundkarte des Computers und die daran angeschlossenen Lautsprecher akustisch wiedergegeben. Während der Reaktionstests, wurde durch einen Timer, die Zeit zwischen dem Präsentieren des Stimulus und Eintreffen der MIDI-Noten ermittelt. Durch Berücksichtigung der Latenz des MIDI-Interfaces, ergibt sich daraus die Reaktionszeit. Die vorgegebene Note wurde mit jeder eintreffenden Note und dem jeweiligen Wert des Timers in eine Textdatei geschrieben.
Für das freie Spielen wurde die Variante des Controllers verwendet, bei der zusätzlich das Excitation Pad gedrückt werden muss, um die ausgewählte Note zu spielen. Zusätzlich zum Auslösen

der Note, konnte durch schräges Drücken der Klang moduliert werden. Außerdem war es möglich eine andere Oktave auszuwählen und über die Druckintensität der Ventiltasten ebenfalls den Klang zu beeinflussen.

3.2.2. Stimuli und akustisches Feedback

Die zu spielende Tonhöhe wurde als Grifftabelle auf dem Monitor angezeigt. Die zum Spielen des Tones benötigten Tasten waren rot und die, von dem Probanden gedrückten, grün eingefärbt.
Bei den Ventiltasten wurden vier übereinander angeordnete Kreise verwendet. Diese sind weiß gefüllt und von einem schwarzen Hintergrund umgeben. Der oberste Kreis war dem Zeigefinger und der unterste dem kleinen Finger zugewiesen. Wenn ein Finger in der Fingerkombination der vorgegebenen Tonhöhe enthalten war, färbte sich der entsprechende Kreis rot (siehe Abbildung 10). Drückte der Versuchsteilnehmer eine oder mehrere Tasten, wechselte die Farbe der entsprechenden Kreise zu grün. Bei der Klaviatur wurde für den Versuch mit zwölf Halbtönen ein Klaviaturausschnitt von einer Oktave von C bis H schematisch angezeigt (siehe Abbildung 11).

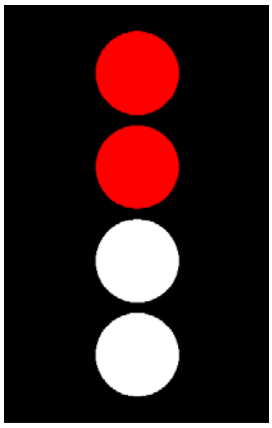

Abbildung 10: Visueller Stimulus für die Ventiltasten

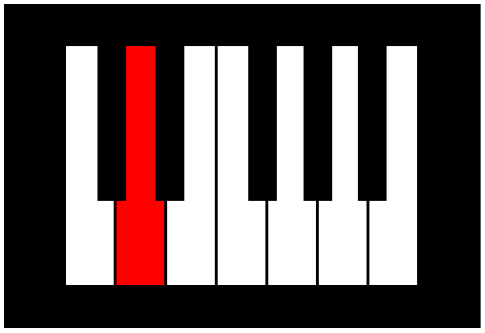

Abbildung 11: Visueller Stimulus für Keyboard

Die weißen Tasten werden als weiße Flächen zu dargestellt. Diese waren von schwarzen Flächen überlagert, welche den schwarzen Tasten entsprechen. Die Größenverhältnisse und die Anordnungen entsprachen der einer Klaviatur. Für den Versuch mit nur vier Tonhöhen standen nur die weißen Tasten von F bis H zur Verfügung. Die zu drückende Taste erschien in rot und die von den Versuchsteilnehmern gedrückte Taste grün.
Als akustisches Feedback wurde die in Kapitel 3.1.6 beschriebene Klangsynthese verwendet. Jedoch gab es außer der Steuerung der Tonhöhe keine variablen Parameter. Diese additive Synthese besteht aus mehreren leicht, zufällig verstimmten Sägezahn Signalgeneratoren. Diese setzten sich aus den ersten 24 Harmonischen der Fourierreihe zusammen. Die Amplitude wird über einer ADSR-Hüllkurve gesteuert. Für den Versuch wurde die Anstiegszeit auf 4 ms gesetzt, um ungewollte Einschaltgeräusche zu vermeiden. Die Ausschwingdauer betrug 100 ms. Das Signal des linken und rechten Kanals passiert jeweils einen Tiefpassfilter, dessen Grenzfrequenz ebenfalls über eine ADSR-Hüllkurve geseteuert wird. Die Anstiegszeit beträgt hier 10 ms und die Ausschwingdauer 200 ms. Nur die Haltedauer der Hüllkurven wurde durch die Versuchsperson gesteuert.
Für jeden Teilversuch wurde ein Pure Data Patch verwendet. Dieses enthielt die entsprechende Stimuliausgabe, die Klangsynthese und die Messung der vergangenen Zeit zwischen Präsentation des Stimulus und eintreffen der MIDI-Note.

3.2.3. Versuchsdurchführung

Die Versuche wurden in dem Zeitraum vom 09.11.2015 bis zum 18.11.2015 durchgeführt. Die Versuchspersonen bekamen jeweils eine ausgedruckte Versuchsanleitung, welche auch die Einwilligungserklärung und die Fragebögen enthielt. Sie wurden gebeten diese vorab durchzulesen. Anschließend konnte mit den Reaktionstests begonnen werden. Diese bestanden aus sechs Teilen. Jeder Versuchsteilnehmer führte den Reaktionstest mit der Klaviatur und dem Controller, mit zwei unterschiedlichen Spielmodi durch. Bei der ersten Variante erfolgte das Spielen des Tones direkt über den Tastendruck und bei der zweiten Variante durch zusätzliches Drücken des Excitation Pads. Mit jeder Interface-Variante wurden zwei Versuchsbedingungen untersucht, eine zwölf und eine mit vier unterschiedlichen Noten. Die Reihenfolge der Reaktionstests war für jede Versuchsperson zufällig. Diese wurde im Vorhinein festgelegt und überprüft, damit sie für jede Versuchsperson unterschiedlich ist.
Die Versuchspersonen wurden instruiert, nach der Anzeige der Tonhöhe, so schnell und fehlerfrei wie möglich den entsprechenden Ton zu spielen. Erst wenn sie den Ton mit der richtigen Tonhöhe gespielt wurde, erfolgt die Anzeige der nächsten. Für jeden Reaktionstest war eine Eingewöhnungsphase über 24 Wiederholungen vorgesehen. Bei dem eigentlichen Versuch wurde jeder Versuchsperson 60 Noten als zufällige Permutationen des Tonhöhenumfangs präsentiert.
Bevor der jeweilige Reaktionstest beginnen konnte, wurde das entsprechende Pure Data Patch geladen und die Nummer der Versuchsperson als Variable eingegeben. Um eine Zuweisung der gespeicherten Daten über den Dateinamen zu ermöglichen. In den MIDI-Einstellung des Programms wurde das verwendete Interface ausgewählt. Nachdem GEM in Pure Data aktiviert wurde, konnte der visuelle Stimulus auf dem externen Monitor, vor der Versuchsperson ausgegeben werden. Nach dem starten des Reaktionstests in Pure Data verließ der Versuchsleiter den Raum und ein Countdown bereitete die Versuchsperson auf den Start des Tests vor. Die Durchführung aller Teilversuche dauerte etwa 30 Minuten. Die Tonhöhenauswahl sollte möglichst mit der dominanten Hand erfolgen. Nach jedem Reaktionstests füllte die Versuchsperson einen kurzen Fragebogen aus und der Versuchsleiter bereitete den nächsten Reaktionstest vor.
Anschließend an die Experimente zur Ermittlung der Reaktionszeiten, hatten die Versuchspersonen 10 Minuten Zeit frei mit dem Controller zu spielen. Dabei war es möglich die in Kapitel 3.1.7 beschriebenen Parameter interaktiv zu steuern. Der Klang wurde dabei, durch die Intensität des Tastendruckes und den Druck auf das Excitation Pad, moduliert.
Abschließend füllten die Versuchspersonen einen Fragebogen zu dem freien Spielen, der Usability des Controllers und ihrem musikalischen Können aus.

3.2.4. Versuchsteilnehmer

Die Probanden wurden über den studentischen E-Mail-Verteiler des Fachgebiets Audiokommunikation rekrutiert. Um weitere Versuchspersonen zu gewinnen, wurden auch ehemalige Studenten und Bekannte angeschrieben. Der Versuch war als Einzeltermin über eine Stunde angesetzt. Um sicher zustellen, dass die Hand ausgewachsen ist, sollten die Probanden älter als 18 Jahre sein. Die musikalische Ausbildung ist kein Auswahlkriterium.
Von den insgesamt 20 Versuchsteilnehmern waren 2 weiblich und 18 männlich. Das durchschnittliche Alter lag bei 28.84 Jahren (SD = 3.20). Die Frage nach der aktuellen Haupttätigkeit ergab, dass 45 % der Teilnehmer studieren und 55 % berufstätig sind. Zusätzlich zu den Fragebögen wurden alle Teilnehmer nach ihrer dominanten Hand befragt, 20 % gaben an, die linke Hand zu bevorzugen und 80 % die rechte.
Unter den Versuchsteilnehmern war kein Berufsmusiker, jedoch gaben alle an, ein Instrument zu

spielen oder gespielt zu haben. 50 % der Teilnehmer haben mindestens ein Jahr und am Höhepunkt des Interesses mehr als eine halbe Stunde am Tag Klavier gespielt. Von den zehn Probanden mit Klaviererfahrung haben nur zwei mehr als drei Jahre das Instrument gespielt und während dieser Zeit mehr als ein Stunde pro Tag geübt.

3.2.5. Auswertungsmethoden

Die Daten der vorgegebenen und gespielten Noten, die vergangene Zeiten von Anzeige bis Eintreffen der MIDI-Note und die Nummer des Durchlaufs wurden für die jeweilige Versuchsbedingung in einer Datei gespeichert. Diese Daten werden zur Auswertung in Matlab eingelesen und deskriptiv analysiert. Von den Reaktionszeiten wird die MIDI-Latenzen des jeweiligen Interfaces subtrahiert (siehe Kapitel 3.2.6).
In einem ersten Schritt werden die arithmetischen Mittelwerte der Reaktionszeiten für die 60 Wiederholungen eines Versuchsdurchlaufes berechnet. Dabei werden die Reaktionszeiten der nicht korrekt gespielten Noten ignoriert. Es ergibt sich für jeden Probanden eine mittlere Reaktionszeit pro Versuchsbedingung.
Zur Bestimmung der Fehlerraten wird für jede der 60 Messwiederholungen überprüft, ob erst eine oder mehrere falsche und anschließend erst die richtige Tonhöhe gespielt wurde. Durch den Vergleich der absoluten Häufigkeit, mit der Anzahl der Wiederholungen, ergibt sich die prozentuale Fehlerrate pro Versuchsteilnehmer und Versuchsbedingung.
Um die arithmetischen Mittelwerte der Reaktionszeiten und Fehlerraten für jede Note zu erhalten werden die Daten nach Noten sortiert. Für die Versuchsbedingungen mit nur vier Noten, ergibt sich unter Berücksichtigung der verwendeten Hand ein fester Fingersatz. Dadurch können die Mittelwerte der Reaktionszeiten und Fehlerraten für jeden Finger bestimmt werden. Für die Versuche mit zwölf Tonhöhen werden diese für jeden Halbton der Oktave ermittelt.
Für die explorative Auswertung der Daten werden diese in SPSS übertragen. Die mittleren Reaktionszeiten und Fehlerraten der Versuchsteilnehmer werden mit der zweifakoriellen Varianzanalyse für wiederholte Messungen untersucht. Der Faktor Interface hat die drei Ausprägungen Klaviatur (KL), Ventiltasten (VT) und Ventiltasten mit drücken oben (VO) und die Anzahl der Noten beträgt vier und zwölf. Es wird untersucht welcher Faktor einen Einfluss auf die unabhängigen Variablen Reaktionszeit und Fehlerrate hat und ob eine Interaktion zwischen ihnen vorliegt. Sofern signifikante Effekte nachgewiesen werden können, wird weiter untersucht, zwischen welchen Bedingungen diese auftreten und in welcher Richtung die Unterschiede gelagert sind. Die Effekte der ANOVA werden durch t-Tests weiter analysiert. Da das Wiederholen des statistischen Test zu einer Kumulation der Alphafehler führt, wird eine Bonferroni-Korrektur des Signifikanzniveaus angewendet.
Die Antworten der Fragebögen auf einer Likert-Skala werden als Zahlenwert in Excel übertragen. Die Texte der Antworten auf offene Fragen werden ebenfalls in die elektronische Form überführt. Bei den offenen Fragen wird das Feedback zu dem Instrument zusammengefasst kategorisiert und nach Häufigkeit der Benennung sortiert. Die am häufigsten genannten Aussagen der Versuchsteilnehmer werden zusammengefasst und wiedergegeben. Es wird dabei auf die Kritikpunkte und Verbesserungsvorschläge eingegangen.

3.2.6. Berücksichtigung der Latenzen

Bei der Auswertung der Reaktionszeiten soll die Latenz zwischen dem Spielen eines Tones und der Registrierung der eingetroffenen MIDI-Note in Pure Data berücksichtigt werden. Wright et al. (2004) nutzen die Aufnahme des Geräusches von dem Tastendruck der Computer-Tastatur auf dem

einen Kanal und dadurch in MAX/MSP ausgelösten Ton auf dem anderen Kanal. Aus der zeitliche Differenz der Audiosignale ergab sich die Verzögerungszeit des interaktiven Systems. Diese Methode fand ihre Anwendung um die MIDI-Latenzen der untersuchten Interfaces zu ermitteln. Das akustische Geräusch, welches das Spielen des Tones mit dem jeweiligen Interface erzeugte, und der in Pure Data generierte Klang wurden aufgenommen. Anschließend konnte anhand der Wellenform der Bereich vom Spielen des Tones bis zum Anfang des Tones in dem Audioeditor Audacity® visuell markiert und die Zeitdifferenz in Millisekunden abgelesen werden. Für das jeweilige Interface wurde jeder Ton einer Oktave fünf Mal gespielt und die entsprechende Verzögerungszeit ermittelt. Die gesamte Latenz des Systems ergab sich durch die Mittlung der Werte. Um nur die MIDI-Latenz zu erhalten wurde die in Pure Data fest eingestellte Audiolatenz von 30 ms von den gemittelten Werten abgezogen.

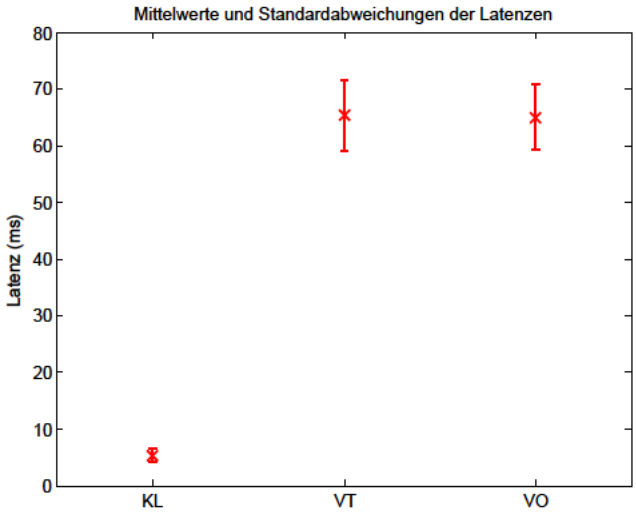

Abbildung 12: Mittelwerte und Standardabweichungen der gemessenen Latenzen, des MIDI-Keyboards (KL), des Controllers mit Ventiltasten (VT) und mit zusätzlichem Drücken des Excitation Pads (V0)

Als Ergebnis ergibt sich ein Mittelwert von 5.31 ms (SD = 1.28) für des MIDI-Keyboards. Für die Variante des Controllers, bei der nur die Tasten gedrückt werden beträgt die ermittelte Latenz 65.28 ms (SD = 6.23) und für Controller bei dem zusätzlich das Excitation Pad gedrückt werden muss 65.01 ms (SD = 5.83). Die Ergebnisse sind in Abbildung 12 dargestellt. Die mittlere Latenz des MIDI-Keyboards ist deutlich geringer als die der beiden Varianten des Controllers.
Die Mittelwerte der Latenzen werden gerundet von den gemessenen Reaktionszeiten des jeweiligen Interfaces abgezogen, um nur die Reaktionszeiten der Versuchsteilnehmer zu erhalten.

4. Ergebnisse

Dieses Kapitel enthält die Evaluationsergebnisse des Instrumentes. Diese ergeben sich aus den Daten, welche mittels der im vorangegangenen Kapitel beschriebenen Methoden erfasst wurden. Das Kapitel ist untergliedert in die Ergebnisse des Reaktionstests und der Fragebögen zum freien Spielen, der Usability des Instrumentes und dem musikalischen Training der Versuchsteilnehmer.

4.1. Ergebnisse des Reaktionstests

Zur Auswertung der Ergebnisse des Reaktionstests werden die Reaktionszeiten und der Fehlerraten untersucht. Für die Reaktionszeiten erfolgt zunächst die graphische Darstellung der Häufigkeitsverteilungen, der unterschiedlichen Versuchsbedingungen. Anschließend folgen die Ergebnisse der Varianzanalyse, anhand derer mögliche Unterschiede zwischen den untersuchten Interface-Varianten aufgedeckt werden sollen. Die Fehlerraten der Versuchsteilnehmer werden ebenfalls mit den explorativen Methoden untersucht. Zur Bestimmung der Signifikanz der Ergebnisse dient SPSS.
Eine Anforderung an den Controller ist, dass nicht einzelne Noten erheblich schwerer zu spielen sein sollten als andere. Um dies zu überprüfen erfolgt abschließend die Bestimmung der mittleren Reaktionszeiten und Fehlerraten pro Note, der jeweiligen Versuchsbedingung. Dabei werden die Ergebnisse aller Versuchsteilnehmer zusammengefasst.
Der folgenden Tabelle sind die verwendeten Bezeichnungen der Untersuchten Versuchsbedingungen zu entnehmen:

Tabelle 5: Übersicht über die Versuchsbedingungen

	Anzahl der Noten	
Interface-Variante	Vier Noten	Zwölf Noten
Keyboard	KL-4	KL-12
Ventiltasten mit direktem Auslösen der Note	VT-4	VT-12
Ventiltasten mit Auslösten der Note über das Excitation Pad oben	VO-4	VO-12

4.1.1. Reaktionszeiten

Bei der Reaktionszeit, handelt sich es um die Zeit von Anzeige der Note, bis zum Spielen der entsprechenden Note. Jedoch wurden bei dem Versuch die Zeiten bis zum eintreffen der MIDI-Noten gespeichert. Zur Ermittlung der Reaktionszeit, wird von diesen Werten die jeweilige MIDI-Latenz des Interfaces aus Kapitel 3.2.6 abgezogen.
Um einen Überblick über die gemessenen Reaktionszeiten der Interface-Varianten mit vier und zwölf Noten zu erhalten, erfolgt zunächst eine graphische Darstellung der Häufigkeitsverteilungen. Die Reaktionszeiten aller Versuchsteilnehmer werden für jede Interface-Variante zusammengefasst und in 40 Gruppen zwischen 200 ms und 2500 ms unterteilt. Alle darüber liegenden Werte, fallen in der Gruppe der höchsten Werte. Die sich ergebenden relativen Häufigkeiten sind in Abbildung 13 dargestellt.

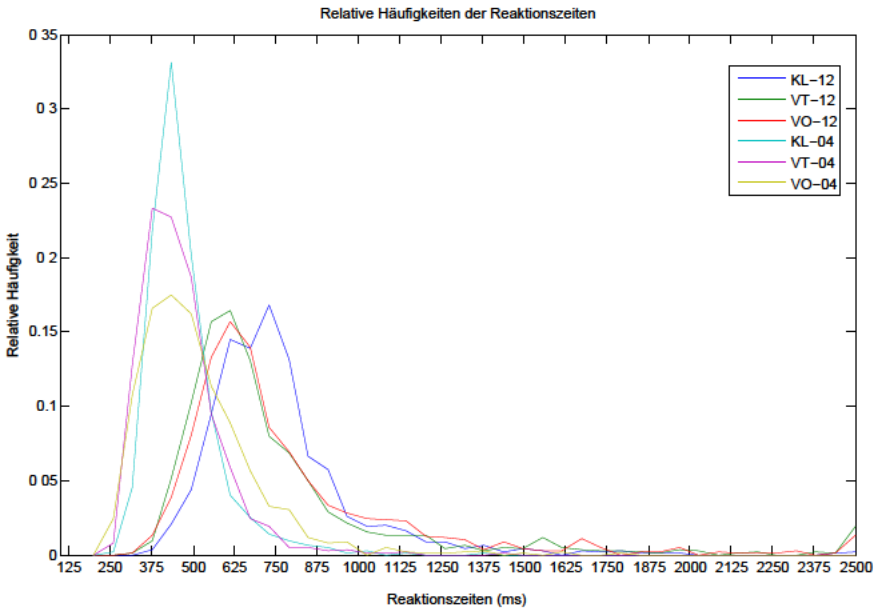

Abbildung 13: Häufigkeitsverteilungen der Reaktionszeiten

Die Betrachtung der Häufigkeitsverteilungen zeigt, dass für die Reaktionszeiten mit festem Fingersatz und nur vier Tonhöhen das Maximum der Verteilungen der unterschiedlichen Interfaces sehr nahe beieinander liegen. Für die Klaviatur ist das Maximum höher als für die beiden Varianten des Controllers. Für den Controller mit Excitation Pad ist das Maximum am geringsten und die Verteilung fällt zu höheren Werten weniger steil ab.
Das Maximum der Häufigkeitsverteilungen für zwölf unterschiedliche Tonhöhen liegt für beide Varianten des Controllers bei etwa 613 ms und bei der Klaviatur etwas höher, bei 731 ms. Die Häufigkeitsverteilung der Klaviatur fällt zu höheren Reaktionszeiten etwas langsamer ab als die des Controllers. Im Gegensatz zur Klaviatur gibt es für die beiden Varianten den Controllers einzelne Reaktionszeiten, welche die obere Grenze der Häufigkeitsverteilung von 2500 ms deutlich überschreiten. Die höchste Reaktionszeit für VT-12 liegt bei 10.1 s und für VO-12 bei 4.9 s. Hingegen liegt der höchste Wert für KL-12 bei 2.9 s.
Die über alle Versuchspersonen gemittelten Reaktionszeiten und die jeweiligen Standardabweichungen für die unterschiedlichen Versuchsbedingungen sind in Tabelle 6 dargestellt.

Tabelle 6: Mittelwerte und Standardabweichungen der Reaktionszeiten in ms

Vier Noten			**Zwölf Noten**		
Interface	M	SD	Interface	M	SD
KL-4	469	48	KL-12	758	101
VT-4	460	53	VT-12	812	352
VO-4	508	109	VO-12	802	217

Der Unterschied zwischen KL-4 und VT-4 ist sehr gering, während die mittlere Reaktionszeit für VO-4 um 48 ms höher ist als für VT-4. Außerdem ist eine höhere Streuung der mittleren

Reaktionszeiten unter den Versuchspersonen für die Variante VO-4 des Controllers zu beobachten. Es zeigt sich bei der Verwendung von zwölf Noten ein sehr geringer Unterschied zwischen den beiden Varianten des Controllers. Hingegen ist die mittlere Reaktionszeit für die Klaviatur etwas geringer als für die beiden Varianten des Controllers.
Die über alle Versuchspersonen gemittelten Reaktionszeiten für die Interface-Varianten der beiden Versuchsbedingungen mit vier und zwölf Noten sind in Abbildung 14 dargestellt.

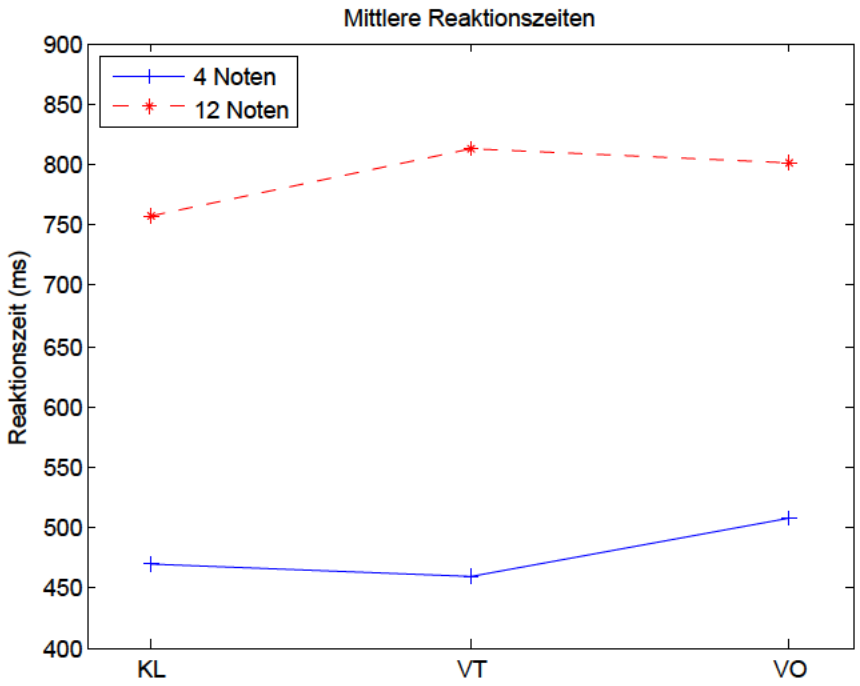

Abbildung 14: Mittelwerte der Reaktionszeiten der Interfaces

Als Voraussetzung für die Varianzanalyse für wiederholtes Messen sollte die Sphärizität gegeben sein. Dazu werden die Varianzen der Messwertdifferenzen mit dem Mauchly-Test untersucht. Als Nullhypothese wird angenommen, dass die Sphärizität gegeben ist. Die Ergebnisse legen nahe die Nullhypothese für die unabhängige Variable Interface $\mathcal{X}^2(2) = 6.44$, $p = .04$ und die Interaktion zwischen Interface und Anzahl der Noten $\mathcal{X}^2(2) = 12.03$, $p = .002$ abzulehnen. Stattdessen wird angenommen, dass die Bedingung der Sphärizität nicht erfüllt ist. Um dies zu kompensieren werden die Freiheitsgrade für die Varianzanalyse nach Greenhouse und Geisser korrigiert ($\varepsilon = .77$ für Interface und .67 für die Interaktion beider Bedingungen).
Die zweifaktorielle Varianzanalyse zeigt, dass der Faktor Interface keinen signifikanten Einfluss auf die Reaktionszeit hat F(1.54, 29.22) = .65, p = .49, η^2= .01. Durch die Varianzanalyse der Reaktionszeiten ist keine Interaktion zwischen den beiden Faktoren Noten und Interface-Variante festzustellen F(1.34, 25.55) = .51, p = .53, η^2= .01. Für den Faktor Anzahl der Noten zeigt sich ein Einfluss auf die Reaktionszeit F(1, 19) = 137.64, p = .00, η^2= .98. Dies ist zu erwarten, da die Entscheidung bei zwölf möglichen Noten die richtige Note auszuwählen mehr Zeit benötigt und der Bewegungsablauf komplexer ist, im Gegensatz zu vier möglichen Noten.

4.1.2. Fehlerraten

Bei den Fehlerraten wird der prozentuale Wert ermittelt. Dieser ergibt sich aus dem Verhältnis der Wiederholungen des Reaktionstests, bei denen nicht auf Anhieb die richtige Note ausgewählt wurde, zu der gesamten Anzahl der Wiederholungen. Dieses Verhältnis wird aus den gespeicherten Daten für jeden Versuchsteilnehmer und die entsprechende Versuchsbedingung ermittelt.
Die über alle Versuchsteilnehmer gemittelten Fehlerraten für die Verwendung von vier möglichen Noten sind in der Tabelle 7 enthalten. Der Mittelwert ist für die Klaviatur etwas geringer als für die beiden Varianten des Controllers, für welche die Mittelwerte kaum voneinander abweichen.
Die Tabelle 8 vergleicht die Mittelwerte und Standardabweichungen der Fehlerraten bei zwölf Noten zwischen den Interface-Varianten. Hier zeigt sich eine deutlich höhere Fehlerrate bei der Verwendung von Fingerkombinationen im Vergleich zur Klaviatur. Dieser Unterschied ist für die Interface-Variante VO-12 etwas geringer als für VT-12. Die mittleren Fehlerraten sind in Abbildung 15 dargestellt.

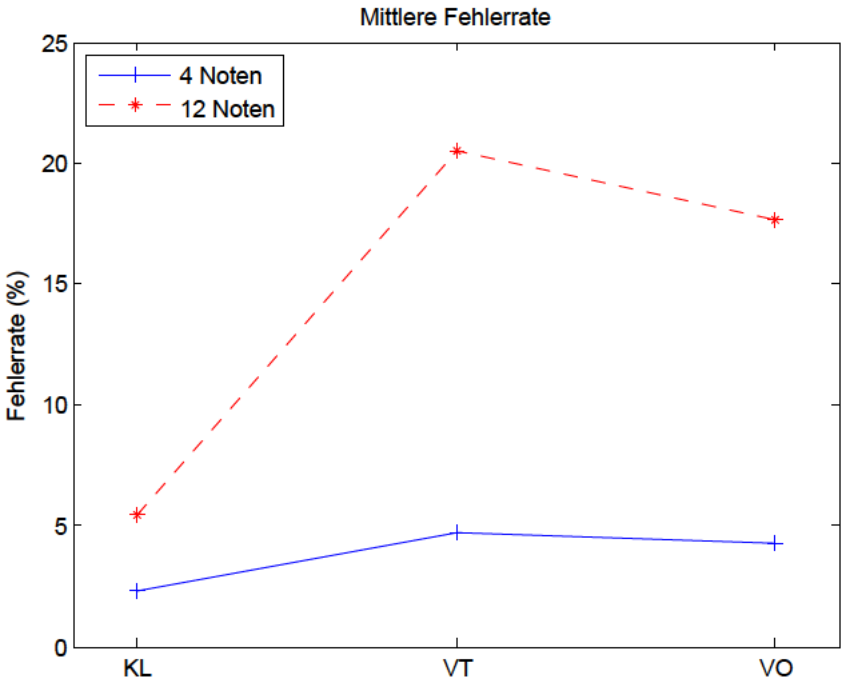

Abbildung 15: Mittelwerte der Fehlerraten der Interfaces

Für die Fehlerrate zeigt der Mauchly-Test, dass für den Faktor Interface $\mathcal{X}^2(2) = 3.10$, $p = .21$ und die Interaktion zwischen Anzahl der Noten und der Interface-Variante $\mathcal{X}^2(2) = 1.02$, $p = .60$, die Annahme der Sphärizität gegeben ist. Dadurch ist es nicht erforderlich die Freiheitsgrade zu korrigieren.
Die zweifaktorielle ANOVA zeigt einen deutlichen Einfluss der Anzahl der Noten und der Interface-Variante auf die Fehlerrate. Sowie eine Interaktion zwischen den beiden Faktoren Anzahl der Noten und Interface-Variante. Für den Haupteffekt Anzahl der Noten ergibt sich ein F-Verhältnis von $F(1, 19) = 98.63$, $p = .00$, $\eta^2 = .57$, für das verwendete Interface $F(2, 38) = 19.26$, $p = .00$, $\eta^2 = .28$ und die Interaktion beider Effekte $F(2, 38) = 12.90$, $p = .00$, $\eta^2 = .15$. Die Effektgröße η^2 ergibt

sich aus der Quadratsumme des jeweiligen Effekts, im Verhältnis zu der gesamten Quadratsumme aller Effekte.
Um weiter zu untersuchen zwischen welchen Interface-Varianten ein signifikanter Unterschied vorliegt und welche Richtung dieser gelagert ist, werden die Fehlerraten jeweils für vier und zwölf Tonhöhen paarweise mit dem t-Test, für abhängige Stichproben verglichen. Da der t-Test für jeweils drei Paare wiederholt wird, muss das Signifikanzniveau von 5 % angepasst werden. Dazu wird die Bonferroni-Korrektur vorgenommen, sodass der α-Fehler von 1.67 % nicht übersteigen darf. Der Shapiro-Wilk-Test weist darauf hin, dass die Messwertdifferenzen der jeweiligen Paare die Voraussetzung der Normalverteilung erfüllen ($p > .05$), außer für die Differenzen der Fehlerraten der beiden Versuchsbedingungen KL-4 und VT-4 ($p = .03$). Da sich der t-Test weitestgehend robust gegenüber leichten Abweichungen von der Normalverteilung verhält (Sawilowsky und Blair, 1992), wird diese Paar nicht von der Untersuchung ausgeschlossen. Die Ergebnisse der paarweisen t-Tests sind in den Tabellen 7 und 8 aufgelistet.

Tabelle 7: Ergebnisse der t-Tests für abhängige Stichproben der mittleren Fehlerraten bei vier Noten, $*p < .017$ (Bonferroni-Korrektur)

Paar	M (%)	SD	df	t	r
KL-4 vs. VT-4	2.33, 4.75	2.98, 4.16	19	−2.93*	.55
KL-4 vs. VO-4	2.33, 4.25	2.98, 3.27	19	−2.93*	.55
VT-4 vs. VO-4	4.75, 4.25	4.16, 3.27	19	.78	.18

Für die Versuchsbedingungen mit nur vier möglichen Tonhöhen und festem Fingersatz zeigen die t-Tests, dass die Fehlerraten für beide Varianten des Controllers im Vergleich zur Klaviatur höher sind. Jedoch sollte die Annahme der Mittelwertgleichheit der Fehlerraten zwischen den beiden Varianten des Controllers nicht verworfen werden.

Tabelle 8: Ergebnisse der t-Tests für abhängige Stichproben der mittleren Fehlerraten bei zwölf Noten, $*p < .017$ (Bonferroni-Korrektur)

Paar	M (%)	SD	df	t	r
KL-12 vs. VT-12	5.42, 20.50	4.08, 9.78	19	−6.23*	.82
KL-12 vs. VO-12	5.24, 17.67	4.08, 10.12	19	−4.80*	.74
VT-12 vs. VO-12	20.50, 17.67	9.78, 10.12	19	.89	.20

Bei Verwendung aller Tonhöhen einer Oktave, ist die mittlere Fehlerrate ebenfalls für beide Varianten des Controllers signifikant höher als für die Klaviatur. Bei dem Vergleich zwischen den beiden Varianten des Controllers, zeigt sich auch hier kein Unterschied.

4.1.3. Vergleich zwischen den Noten

Zur Auswertung der Ergebnisse für die Versuchsbedingungen mit vier Noten, wurden die mittleren Reaktionszeiten für jeden Finger ermittelt. Dazu musste bei der Klaviatur berücksichtigt werden ob die linke oder rechte Hand verwendet wurde. Diese mittleren Reaktionszeiten und Fehlerraten pro Finger sind in den Abbildungen 16 und 17 dargestellt. Es zeigt sich für den Zeigefinger mit allen Interface-Varianten die geringste mittlere Reaktionszeit und Fehlerrate. Die Reaktionszeiten sind für den Mittelfinger am höchsten und fallen über den Ring- zum kleinen Finger ab. Die Reaktionszeiten unterscheiden sich kaum zwischen KL-4 und VT-4, sind jedoch für alle Finger für VO-4 höher.

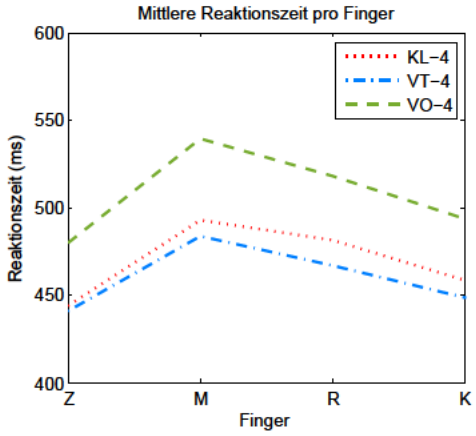

Abbildung 16: Reaktionszeiten je Finger für vier mögliche Noten

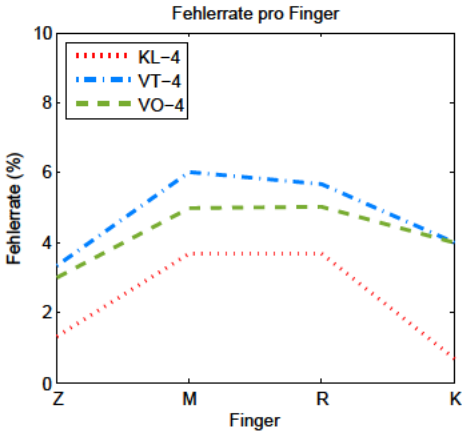

Abbildung 17: Fehlerraten je Finger für vier Noten mit festem Fingersatz

Bei zwölf Tonhöhen ist es nicht möglich nach einem bestimmten Fingersatz auszuwerten. Aus diesem Grund werden die mittleren Reaktionszeiten und Fehlerraten für die jeweilige Note berechnet.

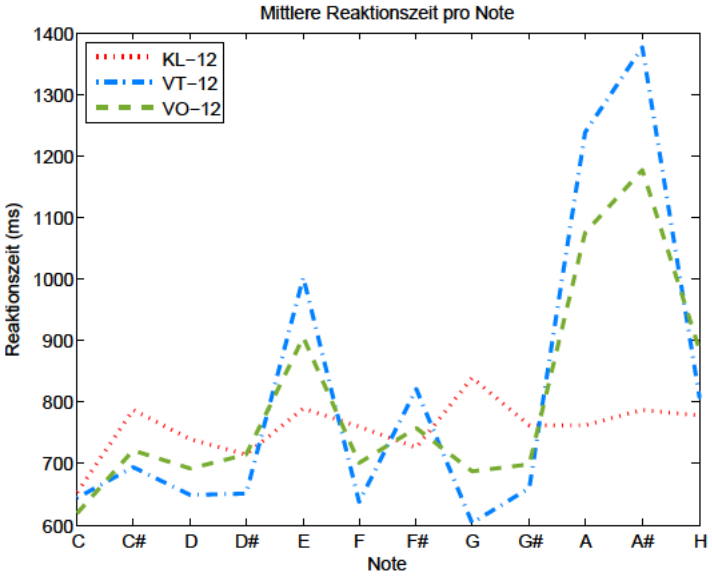

Abbildung 18: Mittlere Reaktionszeiten je Note für zwölf Noten

Die mittleren Reaktionszeiten bei der Klaviatur liegen für alle zwölf Halbtöne zwischen 650 ms für die Note C und 840 ms für das G. Für die beiden Varianten des Controllers sind stärkere Unterschiede zwischen den Noten zu beobachten. Diese liegen für VT-12 zwischen etwa 600 ms für die Note G und 1380 ms bei dem A#. Die Reaktionszeiten sind auch für die Note E mit etwa 1000 ms und A mit 1240 ms auffällig hoch. Für die Interface-Variante VO-12 liegen die Werte zwischen etwa 620 ms für die Note C und 1180 ms für das A#. Wie bei der anderen Variante des Controllers sind auch hier die Reaktionszeiten für die Noten E mit etwa 905 ms und A mit 1075 ms auffällig hoch. In der folgenden Abbildung wird die mittlere Fehlerrate der gesamten Stichprobe der einzelnen Finger für das jeweilige Interface dargestellt.
Die Fehlerrate pro Note zeigt ebenfalls eine Ähnlichkeit zu den Reaktionszeiten pro Note.

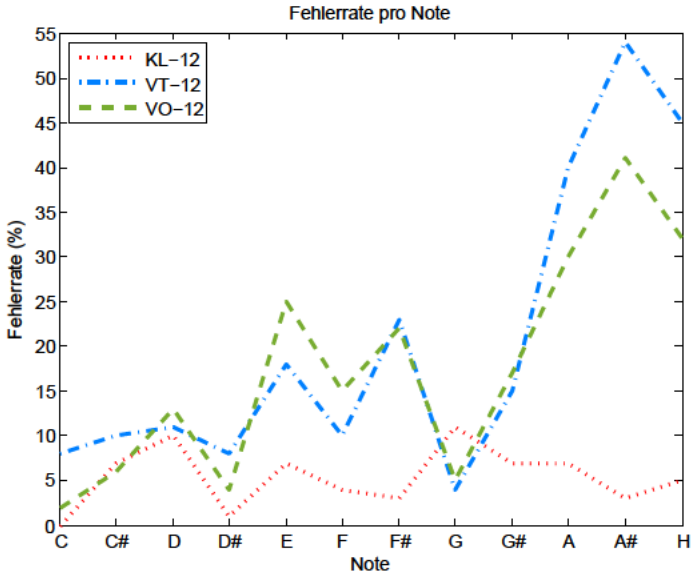

Abbildung 19: Mittlere Fehlerrate je Note für zwölf Noten

Während die Fehlerrate für alle Tonhöhen bei der Klaviatur bei 10 % oder darunter liegen, übersteigen die Fehlerraten von A, A# und H für beide Varianten des Controllers den Werte von 30 %. Nur für die Tonhöhen C, C# D# und G, welche ohne Fingerkombination ausgewählt werden, liegen die Fehlerrate der beiden Varianten des Controllers bei 10 % oder darunter.

4.2. Ergebnisse des freien Spielens

In diesem Abschnitt erfolgt eine deskriptive Auswertung des Fragebogens zum freien Spielen. Die Antworten zu den abgefragten Items des freien Spielens wurden auf einer Likert-Skala von 1 bis 5 angegeben. Die Gruppenmittelwerte und Standardabweichungen werden ermittelt und als Wertung interpretiert.
Es zeigt sich, dass die Versuchsteilnehmer das Spielen mit dem Controller als inspirierend empfinden. Nach Aussage der Versuchspersonen, ist die Interaktion mit der Klangsynthese direkt und der Controller reagiert schnell auf die Gesten. Die musikalischen Möglichkeiten werden als umfangreich eingestuft und die Mehrheit der Versuchsteilnehmer würde, wenn in Zukunft die Möglichkeit bestünde, das Instrument noch einmal verwenden. Jedoch war den Versuchsteilnehmern das Spielen mit dem Instrument nur wenig vertraut. Das Gefallen des Klanges ist neutral.
Die Probanden hatten am Ende des Fragebogens die Möglichkeit in freier Textform weiteres Feed-

back zu dem Instrument zu geben. Daraus ergeben sich viele sehr nützliches Verbesserungsvorschläge, positives Feedback und Kritik.

Tabelle 9: Fragen zu dem freien Spielen und die entsprechenden arithmetischen Mittelwerte über alle Versuchsteilnehmer und die Standardabweichungen der Antworten

	Frage	M	SD
1.	Wie aufregend empfanden Sie das Spielen des Instrumentes?	3.90	0.64
2.	Wie inspirierend empfanden Sie das Spielen des Instrumentes?	4.00	0.73
3.	Wie fließend empfanden Sie das Spielen des Instrumentes?	3.30	1.03
4.	Wie vertraut empfanden Sie das Spielen des Instrumentes?	2.15	0.88
5.	Wie direkt empfanden Sie die Interaktion?	4.00	0.79
6.	Wie geeignet finden Sie das Instrument?	3.85	0.75
7.	Wie schnell hat das Instrument auf Ihre Gesten reagiert?	4.40	0.75
8.	Wie genau hat das Instrument auf Ihre Gesten reagiert?	3.60	1.05
9.	Als wie anwenderfreundlich empfanden Sie das Instrument?	3.50	0.69
10.	Wie umfangreich schätzen Sie die musikalischen Möglichkeiten des Instrumentes ein?	4.10	0.91
11.	Wie schnell haben Sie Fortschritte beim Erlernen gemacht?	3.45	0.94
12.	Wenn Sie die Möglichkeit hätten, würden Sie das Instrument in Zukunft noch einmal verwenden?	4.15	0.81
13.	Wie gut gefällt Ihnen der Klang des Instrumentes?	3.47	1.02

Die wesentlichen Aussagen der Teilnehmer werden in dem folgenden Abschnitt zusammengefasst. Die Steuerung der Grenzfrequenz und des Q-Faktors durch schräges Drücken, auf die obere Fläche, wird von drei Teilnehmern als sehr empfindlich beschrieben. Sie stellen fest, dass es ein Vorteil wäre, wenn diese Parameter präziser steuerbar wären. Ein Vorschlag beinhaltete, dass diese Steuerung deaktivierbar sein sollte. Eine Anmerkung war, dass stark gedrückt werden muss, um die volle Lautstärke zu erhalten.
Drei der Teilnehmer haben den Eindruck, dass auch die Ventiltasten zu fest gedrückt werden müssen. Dies empfindet eine Versuchsperson als „anstrengend“ und ein anderer Proband merkt an, dass „leichtes Drücken ... angenehm“ wäre. Es wird der Vorschlag gemacht, die Ventiltasten weniger abstehend zu gestalten, um diese leichter drücken zu können. Ein Proband findet die Ventiltasten hingegen „sehr gut'". Die Tastenkombinationen werden von einer Versuchsperson als „teilweise nicht sehr intuitiv“ beschrieben und ein Teilnehmer hat „Schwierigkeiten bei den Übergängen zwischen den Tönen“.
Die Oktavauswahl mit dem Daumen der gleichen Hand, empfindet ein Teilnehmer als „schwer“ zu bedienen. Es wird vorgeschlagen die Schalter zur Oktavauswahl etwas höher anzubringen, damit die höchste Oktave einfacher für den Daumen zu erreichen ist. Es wird angemerkt, dass die „Koordination der Hände ... nicht gerade leicht“ sei, „aber sicher erlernbar“. Das Feedback mancher Teilnehmer bezieht sich auch auf den Klang. Ein Teilnehmer stellt fest: „Distortion wäre gut“, ein anderer wünscht sich „weniger LFO“ Modulation. Es werden die Vorschläge gemacht die Ausklingzeit als steuerbaren Parameter anzubieten, oder eine „Loop Funktion“ zu integrieren. Einem Teilnehmer fällt auf, dass es vereinzelt zu unerwünschten Knacken kommen kann, wenn die Puffergröße der Soundkarte nicht ausreicht.
Die Spielweise ist für einzelne schwierig, da der Klang bei der Anregung kontinuierlich moduliert wird. Dadurch sei es schwer Melodien mit einem gleichbleibenden Klang zu spielen. Einem anderen gefällt hingegen die Modulationsmöglichkeit „sehr gut“. Manche Versuchsteilnehmer spie-

len dadurch eher weniger Melodien, sondern nutzen eher die Modulationsmöglichkeiten um den Klang zu steuern oder wie ein Teilnehmer schreibt, um „Klangteppiche“ zu erzeugen.

4.3. Ergebnisse des System Usability Scores

Zur Berechnung des SUS-Score, werden von den Bewertungen der Aussagen mit ungerader Nummer, eins abgezogen und die mit ungerader von fünf abgezogen. Dadurch erhalten alle Bewertungen die gleiche Polarität. Anschließend werden diese für jede Versuchsperson aufsummiert und mit 2.5 multipliziert. Daraus ergibt sich der SUS-Score, der zwischen 0 und 100 liegen kann. Aus den Bewertungen des erprobten Instrumentes zu den Aussagen der SUS, ergibt sich ein mittlerer Score von 65 (SD = 11.58). Die Aussagen, die arithmetischen Mittelwerte und Standardabweichungen der Bewertungen der SUS, sind in der Tabelle 10 dargestellt.

Tabelle 10: Für das Instrument angepasste Aussagen der System Usability Scale, arithmetischer Mittelwert und Standardabweichung der Bewertungen

	Aussage	M	SD
1.	Ich denke, ich würde das Instrument gerne weiterhin spielen.	3.70	0.57
2.	Ich finde das Instrument unnötig kompliziert.	2.10	0.79
3.	Ich denke das Instrument war einfach anzuwenden.	3.45	0.89
4.	Ich würde die Unterstützung eines Lehrers benötigen um damit Spielen zu lernen.	2.10	1.21
5.	Die Eingabemöglichkeiten sind gut in das Instrument integriert.	4.00	0.56
6.	Es gab zu viele Unstimmigkeiten bei der Anwendung.	2.10	0.79
7.	Ich könnte mir vorstellen, dass die meisten Menschen das Instrument schnell erlernen können.	3.50	0.89
8.	Ich finde, die Bedienung des Instrumentes war umständlich.	2.30	0.73
9.	Ich fühle mich mit der Anwendung des Instrumentes sehr vertraut.	2.85	0.99
10.	Ich muss noch eine Menge lernen bis ich mit dem Spielen des Instrumentes fortfahren kann.	3.00	1.41

Auf dem Fragebogen der SUS hatten die Versuchspersonen die Möglichkeit in freier Textform anzugeben, welche Art von Problemen oder Fehlbedienungen bei der Erprobung aufgetreten sind und aus welchem Grunde Sie das Instrument für geeignet halten.
Fünf Teilnehmer geben an, dass es während der Erprobung zu Problemen oder Fehlbedienungen gekommen ist. Bei einem Teilnehmer reagiert das Programm nicht mehr auf die Gesten und die USB-Verbindung musste unterbrochen und wieder hergestellt werden, um das Problem zu lösen. Ein weiterer gibt an, dass es zu ungewollten Zwischentönen kommen kann, wenn nicht alle Tasten für die gewünschte Tonhöhe gleichzeitig gedrückt werden. Für einen weiteren reagieren die Tasten zu sensibel und zu schnell im Verhältnis zu der Kraft, die aufgewendet werden muss. Zwei weitere Teilnehmer stellen fest, dass es in manchen Fällen Probleme beim Auslösen des Tones gibt und es vorkommen kann, dass kein Ton gespielt wird, obwohl das Excitation Pad gedrückt wird.
Es werden unterschiedlichste Gründe angeführt, warum die Teilnehmer das Instrument für geeignet halten. Beispielsweise, da es „flexibel einsetzbar“ ist, es „viel Dynamik ins Spielen bringen kann“, oder es über „gute Modulationsmöglichkeiten“ verfügt. Die einfache oder „intuitive Bedienung“ wird mehrmals als Grund angegeben. Ein Teilnehmer ist der Meinung, dass dadurch bereits Anfänger „interessante und lebendige Sounds“ erzeugen können. Andere führen Gründe an, wie das Training der „Fingerkoordination und Muskulatur“, das Instrument „gut in der Hand

zu halten“ oder die Größe vorteilhaft ist. „Die alternative Bedienung fasziniert und bietet neue Möglichkeiten in der Klanggestaltung“.

5. Diskussion

In diesem Kapitel werden die Ergebnisse der Untersuchung, bestehend aus dem Reaktionstest und dem freiem Spielen diskutiert und kritisch betrachtet. Daraufhin wird der Realisierungprozess und der daraus entstandene Prototyp diskutiert. Aus dem Feedback der Fragebögen und den Erfahrungen während des Entwicklungsprozesses werden Verbesserungsmöglichkeiten abgeleitet. Den Abschluss bildet ein Ausblick mit Überlegungen zur Erweiterung des Controllers. Dabei werden alternative Einsatzmöglichkeiten besprochen.

5.1. Reaktionstests

Für die Versuchsbedingung mit vier Noten ist die Spielweise der Klaviatur und der Ventiltasten weitestgehend identisch. Die Finger sind immer fest den Tasten zugeordnet und es werden keine Fingerkombinationen verwendet. Allerdings sind die Tasten bei den beiden Varianten des Controllers vertikal und bei der Klaviatur horizontal angeordnet. Der Unterschied zwischen den beiden Varianten des Controllers, ist das zusätzliche Drücken des Excitation Pads, um die Note auszulösen. Für die Reaktionszeit zeigte sich kein signifikanter Einfluss der Interface-Variante. Daraus lässt sich die Annahme ableiten, dass die unterschiedliche Mechanik der Tasten keinen Einfluss auf die Reaktionszeit hat. Diese ist für die Variante des Controllers, bei der die Note durch das Excitation Pad ausgelöst wird, im Durchschnitt 48 ms höher als für das direkte Auslösen. Bei der Fehlerrate zeigt sich ein geringer, jedoch signifikanter Unterschied zwischen der Klaviatur und den beiden Varianten des Controllers. Dies könnte auf die vertikale Anordnung der Tasten zurückzuführen sein. Was eventuell zu einer schwereren Zuordnung des Stimulus zu dem jeweiligen Finger führt.
Bei der Verwendung von zwölf Noten, löst sich für die Klaviatur der feste Fingersatz auf und bei den beiden Varianten des Controllers kommen Fingerkombinationen zum Einsatz. Dies führt in beiden Fällen zu deutlich höheren Reaktionszeiten. Jedoch führt dies nicht zu signifikanten Unterschieden in der Reaktionszeit zwischen den Interface-Varianten. Die Fehlerrate ist für die beiden Varianten des Controllers deutlich höher als für die Klaviatur. Mögliche Ursachen dafür sind, dass die Versuchspersonen Schwierigkeiten hatten die benötigten Tasten simultan zu drücken, die roten Kreise des visuellen Stimulus den Fingern zuzuordnen oder die Bewegung auszuführen. Die hohe Anzahl der Fehler bei den Fingerkombinationen im Vergleich zur Klaviatur kann durch das zusätzliche Drücken des Excitation Pads von 21 % auf 18 % verringert werden. Jedoch zeigt der Vergleich durch den t-Test, dass dieser Unterschied nicht signifikant ist. Bei der Betrachtung der durchschnittlichen Fehlerraten der einzelnen Noten, zeigt sich, dass diese bei gewissen Noten, wie dem A, A# oder H, deutlich höher sind als bei der Klaviatur. Dies lässt vermuten, dass die Bewegungen für manche Fingerkombinationen besonders schwer auszuführen sind. Allgemein unterscheiden sich die Fehlerraten zwischen den Fingerkombinationen deutlich. Dies weist darauf hin, dass manche Kombinationen besser geeignet sind als andere.
Wenn bei den Fingerkombinationen nicht alle Finger zeitgleich gedrückt werden, wird für kurze Zeit eine andere Tonhöhe ausgewählt. Gegebenenfalls ist das simultane Drücken mehrerer Tasten durch Übung erlernbar. Die hohe Fehlerrate weist darauf hin, dass dies für Anfänger im Rahmen des Reaktionstests jedoch schwierig ist. Um diese Art der Fehler zu vermeiden, wäre eine Verlängerung des vorgesehenen Zeitfensters von 46 ms denkbar. Die Folge ist eine zusätzliche Vergrößerung der Latenz. Jedoch ist diese mit etwa 65 ms bereits zu hoch. Alternativ könnte das Zeitfenster für den Nutzer variabel einstellbar sein, um Anfängern den Einstieg zu erleichtern.
Der Reaktionstest alleine lässt keine Aussage über das Instrument als Ganzes zu. Jedoch ist er zum Vergleich der Tonhöhensteuerung über Vier-Finger-Kombinationen mit der Klaviatur ein nützli-

ches Mittel, um trotz der wesentlichen Unterschiede der Interfaces und der Bewegungsabläufe aussagekräftige Ergebnisse zu erzielen. Die Verwendung von Vier-Finger-Kombinationen zur Tonhöhensteuerung führt zu keinem Unterschied in der Reaktionszeit jedoch zu einer höheren Fehlerrate im Vergleich zur Klaviatur.

5.1.1. Fragebogen zum freien Spielen

Ein Zeitraum von etwa zehn Minuten ist recht kurz, um sich mit einem digitalen Musikinstrument intensiv auseinanderzusetzen und sich mit dessen Spielweise vertraut zu machen. Das freie Spielen bietet die Möglichkeit den Funktionsumfang auszuloten und die Steuerung und den Klang des Instrumentes kennenzulernen. Auch wenn der Funktionsumfang des untersuchten Prototyp des Instrumentes noch überschaubar ist, wurden die musikalischen Möglichkeiten von den Versuchsteilnehmern als umfangreich eingeschätzt. In dieser kurzen Zeit können weitestgehend alle Spielweisen erprobt werden. Jedoch ist sie nicht ausreichend das Mapping der Tonhöhensteuerung zu erlernen und komplexere Melodien zu spielen. Dadurch beziehen sich die Aussagen und das Feedback stärker auf die Modulation des Klanges, als auf die Tonhöhensteuerung über Vier-Finger-Kombinationen.
Der Fragebogen zum freien Spielen dient dazu das Instrument zu Bewerten und eine Auskunft über die Erfahrungen während des Spielens zu geben. Das Feedback zu dem Instrument, das die Versuchsteilnehmer offen formulieren konnten ist sehr umfangreich und beleuchtet unterschiedlichste Aspekte. Dadurch ergeben sich viele wertvolle Verbesserungsvorschläge, durch die das Instrument optimiert werden kann. Beispielsweise sollte die Klangmodulation über das Excitation Pad präziser sein, die Anordnung der Oktavschalter etwas höher oder die Ventiltasten etwas weniger weit nach vorne abstehend. Da die Gestaltung der Klangsynthese sehr zeitaufwendig ist und nicht Hauptfokus dieser Arbeit, zeigte sich bei der Evaluation, dass der Klang eher neutral bewertet wurde.

5.1.2. System Usability Scale

Nach Bangor et al. (2008) erzielen akzeptable Produkte einen SUS-Score über 70, gute Produkte einen Werte um die 80 und überragende Produkten einen Wert von über 90 Punkten. Ist der Score jedoch unter 70, sollten Verbesserungen vorgenommen werden. Aus den Bewertungen des Prototypen durch die SUS ergibt sich ein durchschnittlicher Score von 65 (SD = 11.58). Dies zeigt, dass der realisierte Prototyp noch keinen finalen Stand erreicht hat und noch Raum für Verbesserungen bietet. Bei genauerer Betrachtung der Bewertungen zeigt sich, dass der geringe SUS-Score hauptsächlich auf den hoch eingestuften Lernaufwand und die geringe Vertrautheit mit dem Instrument zurückzuführen ist. Dies könnte wiederum mit der kurzen Nutzungsdauer während des Versuches in Zusammenhang stehen. Möglicherweise erhält das Instrument eine höhere Bewertung, wenn die Befragten mehr Zeit hätten, auf dem Instrument zu spielen und sich dadurch mit der Art der Tonhöhensteuerung besser vertraut machen könnten.
Allgemein stellt sich die Frage, ob die SUS ein geeignetes Mittel ist, um digitale Musikinstrumente zu bewerten. Diese ist eine sehr schnelle und vereinfachte Methode zur Bewertung von Systemen. Die SUS liefert nur eine sehr grobe und oberflächliche Einschätzung der Usability und lässt keine Aussage über die Expressivität eines Instrumentes oder den kreativen Prozess des Musizierens zu. Daher ist die Antwort auf die Frage abhängig von dem Ziel der Untersuchung. Da hier nur eine einfache Einschätzung der Versuchsteilnehmer erzielt werden sollte, ist die modifizierte Variante der SUS für die Untersuchung im Rahmen dieser Arbeit hilfreich. Da der Fragebogen ebenfalls

die Möglichkeit bietet auf Probleme und Fehlbedienungen hinzuweisen, ergeben sich daraus Hinweise, wie das System optimiert werden kann.

5.2. Evaluationsmethode

Bei der Evaluation eines digitalen Musikinstrumentes können unterschiedliche Perspektiven eingenommen werden (O'modhrain, 2011). Je nach Sichtweise ergeben sich unterschiedliche Kriterien. Für ein Musiker wird es von großer Bedeutung sein, wie schwer das Instrument zu erlernen ist, welche Ausdrucksmöglichkeiten und klangliche Vielfalt es ihm bietet. Für den Designer steht im Vordergrund, ob das Instrument die Anforderungen und die Ziele der Usability erfüllt. Hingegen spielt es für das Publikum eine wichtige Rolle wie interessant und aufregend die musikalische Performance ist. Allerdings ist dies nicht nur von dem Instrument sondern auch von dem Können des Musikers abhängig. Die Evaluation des realisierten Controllers fokussiert sich auf die Sichtweise des Musikers und des Designers. Für den Musiker ist es von Bedeutung ob die Noten schnell und fehlerfrei gespielt werden können. Hingegen ist es für den Designer wichtig ein Feedback zu der Usability und Verbesserungsmöglichkeiten zu bekommen.
Bei dem durchgeführten Experiment wurde die Art der Tonhöhensteuerung untersucht. Die Kriterien beschränken sich dabei auf die benötigte Zeit, um eine Note zu spielen und die Fehlerrate. Die Übergänge von einer Note zur nächsten Note wurde dabei nicht betrachtet. Bei dem Controller müssen je nach Intervall und Tonhöhe mehrere Finger umgreifen. Die Lernkurven von Bewegungsabläufen können sehr unterschiedlich verlaufen (Anderson, 2013, S. 5-7). Für die Erlernbarkeit der Spielweise sind die Fortschritte durch regelmäßiges Üben besonders wichtig. Um diese zu untersuchen sind noch weiter Experimente notwendig.
Durch die Verwendung von maximal vereinfachten Aufgaben verringert sich die externe Validität der Ergebnisse, zugunsten der internen Validität. Um dies auszugleichen, hatten die Teilnehmer die Möglichkeit frei mit dem Instrument zu spielen. Durch die kurze Zeitdauer des freien Spielens haben die Teilnehmer einen ersten Eindruck von dem Instrument bekommen, der sich in den Antworten der Fragebögen widerspiegelt. Für eine umfangreichere Evaluation wäre es von Interesse, dass das Instrument über einen Zeitraum von einem oder mehreren Musikern erlernt wird, um umfangreichere Erfahrungen in die Evaluation einfließen zu lassen. In einem weiteren Schritt könnte auch das Publikum nach einer Aufführung befragt werden, um das Instrument aus unterschiedlicheren Perspektiven zu evaluieren.

5.3. Realisierung des Controllers

Das Resultat des realisierten Controllers ist ein Instrument das umfangreiche musikalische Möglichkeiten bietet. Neben der Tonhöhensteuerung können über den Druck der Ventiltasten Parameter der Klangsynthese gesteuert werden. Das Triggern von Noten ist verbunden mit der Gewinnung von weiteren Sensordaten, die ebenfalls auf Parameter der Klangsynthese abgebildet werden können. Durch das kontinuierliche Anwenden und Testen des Controllers, wurde die Auswertung der Sensordaten optimiert. Dies ist ein sehr zeitaufwändiger Prozess, der jedoch notwendig ist um das Spielgefühl zu optimieren.
Das Mapping der Fingerkombinationen auf die Tonhöhe hat sich aus dem Konzept des Controllers heraus ergeben. Alternative Mappings sind jedoch möglich. Beispielsweise könnte die Zuweisung dem Ansatz von Beauregard (1991) folgen und Gray-Codes verwenden (siehe Kapitel 2.1.2). Dadurch müsste nur ein Finger wechseln, um die Tonhöhe einen Halbtonschritt zu erhöhen oder zu verringern. Allerdings würde dadurch der logische Zusammenhang der binären Zuweisung verloren gehen. Es müsste jede Kombination erlernt werden und wäre nicht über den Zusammenhang

herleitbar. Eine weitere Alternative, ist die Tonhöhenauswahl entsprechend eines Blechblasinstrumentes mit Quartventil zu gestalten. Der Grundton des Instrumentes wird gespielt, wenn keine Taste gedrückt ist. Durch Drücken des ersten Ventils wird die Tonhöhe um zwei Halbtöne, des zweiten um einen Halbton, des dritten um eine Terz und des vierten Ventils um eine Quarte abgesenkt. Dadurch entsteht neben dem logischen Zusammenhang auch ein musikalischer, da sich die Tonhöhe durch Wechseln einzelner Finger in vorgegebenen Intervallen ändert. Allerdings könnten dadurch nur elf Halbtöne ausgewählt werden, sodass in dem Zustand in dem keine Taste gedrückt ist, der Grundton des Instrumentes gespielt wird. Gleiches gilt für die Verwendung von Gray-Codes.
Eine Besonderheit des Controllers ist das Excitation Pad. Bei dessen Betätigung wird einerseits die Note gespielt und gleichzeitig der Klang durch schrägen Druck beeinflusst. Dadurch sind die Anregung und die Modulation nicht unabhängig. Wenn die Steuerung nicht ausreichend präzise ist, kann beim Spielen einer Melodie der Klang unter Umständen ungewollt moduliert werden. Dies muss beim Spielen beachtet werden und erfordert entsprechendes Feingefühl.
Von den Versuchsteilnehmern wurde die Latenz des Controllers nicht bemängelt. Jedoch zeigen die Messungen, dass die Zeit vom Auslösen der Note bis zum akustischen Feedback mit etwa 95 ms deutlich zu hoch ist. Ein Grund für die hohe Latenz des Controllers ist das Zeitfenster, innerhalb dessen alle Tasten zur Tonhöhenauswahl gedrückt werden müssen, um ungewollte Zwischentöne zu vermeiden. Weitere Gründe sind die Latenz der USB-MIDI-Schnittstelle des Teensy Boards und der Soundkarte.
Durch das interdisziplinäre Team konnten während der Entwicklung mögliche Probleme erkannt und vorgebeugt werden. Die Evaluation und das Feedback der Versuchsteilnehmer zeigen noch weitere Verbesserungsmöglichkeiten auf. Durch die verwendeten Silikonscheiben als Puffer für die Tasten muss ein Druck von etwa 8 N aufgebracht werden, um den maximalen Wert zu erreichen. Es wird sich zeigen, ob das Silikon über längere Zeit beständig bleibt.
Die entworfene Klangsynthese ist nur eine von vielen Möglichkeiten den Controller einzusetzen. Durch die einfache Struktur bietet sie nur eingeschränkte Modulationsmöglichkeiten. Durch die Verwendung bestehender Lösungen, kann der Fokus auf die Interaktion und das Mapping der Steuerparameter gerichtet werden.
Da für die Anwendung der Klang und das Mapping von großer Bedeutung sind, ist es möglich, großen Aufwand zu betreiben um diese Komponenten optimal aufeinander abzustimmen. Dies ist bei digitalen Musikinstrumenten häufig ein kontinuierlicher Prozess, in dem sich aus der Benutzung Verbesserungen ergeben, auf deren Basis iterativ Anpassungen vorgenommen werden. Experimente mit Pure Data Patches und bestehenden Software-Synthesizern liefern bereits vielversprechende Ergebnisse.

5.4. Tonhöhensteuerung über Vier-Finger-Kombinationen

Die Evaluation der Tonhöhensteuerung über Vier-Finger-Kombinationen zeigt, dass diese eine Alternative zur Klaviatur darstellt. Jedoch birgt sie das Risiko, dass für kurze Zeit ungewollt ein Zwischenton gespielt oder nicht die richtige Tonhöhe ausgewählt wird. Simultanes Drücken von mehreren Tasten bedarf anfangs viel Übung. Ebenso wie das Erlernen von mechanischen Musikinstrumenten ein langer Prozess ist.
Eventuell kann die Tonhöhensteuerung durch ein anderes Mapping als die Binär-Codes optimiert werden. Eine mögliche Alternative sind die Tastenkombinationen eines Blechblasinstrumentes mit Quartventil. Der größte Vorteil wäre dabei, dass alle Musiker, welche diesen Fingersatz bereits beherrschen, das Instrument mit deutlich geringerem Aufwand erlernen könnten. Oder es wäre die Verwendung der Gray-Codes wie beim Blauregard (Beauregard, 1991) denkbar. Allerdings wird

bei diesen beiden Varianten immer der Grundton gespielt, wenn keine Taste mehr gedrückt ist. Beim implementierten Mapping wird die letzte ausgewählte Tonhöhe gehalten, bis das Excitation Pad nicht mehr gedrückt wird. Dadurch kann der Ton gespielt und moduliert werden, während die Finger für die nächste Fingerkombination angeordnet werden.
Durch den Reaktionstest zeigt sich, dass manche Fingerkombinationen schwerer zu greifen sind als andere. Beispielsweise sind die Reaktionszeit und Fehlerrate für die Kombination aus Mittel- und kleinem Finger deutlich höher. Eine separate Untersuchung wäre notwendig, um die Reaktionszeiten und Fehlerraten aller möglichen Fingerkombinationen zu bestimmen. Die mit den besten Ergebnissen könnten bei dem Mapping bevorzugt werden. Bei dem Spielen von Melodien sind auch die Übergänge zwischen den einzelnen Tönen wichtig. Für den Vergleich unterschiedlicher Mappings führt dies zu komplexen Aufgaben bei der Evaluation.

5.5. Ausblick

Durch die Realisierung des Controllers wurde die Grundlage geschaffen, um Tonhöhen auszuwählen und simultan Klangparameter zu steuern. Neben der Tonhöhensteuerung über Binär-Codes, lassen sich weitere Mapping-Strategien implementieren und evaluieren. Für den Verglich unterschiedlicher Mappings wird es notwendig sein, dass diese über einen längeren Zeitraum erlernt werden. Über gut ausgewählte und komplexere Aufgaben wäre eine Bewertung möglich.
Der Controller benötigt noch einen offiziellen Namen. In Anlehnung an das binäre Mapping wurde der vorläufige Arbeitstitel BinBong gewählt.
Die Hardware des Controllers ist erweiterbar. Die Elektronik kann mit zusätzlichen Sensoren, zur Erkennung des Neigungswinkels oder der Beschleunigung aufgerüstet werden. Neben der direkten Tonhöhensteuerung ist auch eine Navigation in einem Tonraum, eine interaktive Steuerung von Sequenzen oder das Spielen von Harmonien denkbar.
Durch die Implementierung einer Spectral Modeling Synthesis (SMS) wäre es möglich, den Klang des Instrumentes komplexer und reichhaltiger zu gestalteten. Zur Steuerung sollten alle Sensordaten direkt in Echtzeit an den Computer übertragen werden. Dadurch könnte das Mapping flexibler gestaltet und eine höhere Auflösung der Steuerdaten erreicht werden.
Es wäre sehr erfreulich wenn weitere Musiker die Möglichkeit bekommen das Instrument zu spielen und durch ihre Erfahrungen zu einer weiteren Optimierung beitragen könnten. Dies wäre ein weiterer Schritt in der Entwicklung von digitalen Musikinstrumenten.

Abbildungsverzeichnis

1. Akai EWI USB Wind-Controller, Quelle: Akai Professional (2015) 6
2. Zeichnung des alternativen Wind-Controllers, Quelle: Beauregard (1991) 7
3. Skizzen der unterschiedlichen Konzepte 25
4. Acrylplatte links, Sensoren mittig und Acrylrohr rechts 26
5. Ventiltasten, welche mit dem Acrylrohr verschraubt sind 27
6. Piezoschalter mit der Messingplatte 28
7. Silikonpuffer und Sensoren zur Steuerung der Anregung 29
8. Teensy-Board erweitert um die Sensoren und notwendigen Bauteile 30
9. Versuchsaufbau 38
10. Visueller Stimulus für die Ventiltasten 39
11. Visueller Stimulus für Keyboard 39
12. Mittelwerte und Standardabweichungen der gemessenen Latenzen, des MIDI-Keyboards (KL), des Controllers mit Ventiltasten (VT) und mit zusätzlichem Drücken des Excitation Pads (V0) 42
13. Häufigkeitsverteilungen der Reaktionszeiten 44
14. Mittelwerte der Reaktionszeiten der Interfaces 45
15. Mittelwerte der Fehlerraten der Interfaces 46
16. Reaktionszeiten je Finger für vier mögliche Noten 48
17. Fehlerraten je Finger für vier Noten mit festem Fingersatz 48
18. Mittlere Reaktionszeiten je Note für zwölf Noten 48
19. Mittlere Fehlerrate je Note für zwölf Noten 49
20. Übersicht über verwendete Materialien, dargestellt im Querschnitt des Controllers 71
21. Seitenansicht des Controllers 72
22. Ansicht der Rückseite des Controllers 73
23. Programmablaufplan der Firmware mit MIDI Implementierung 74

Tabellenverzeichnis

1. EVI-Fingersatz nach Akai Professional (2015). Ein gefüllter Kreis bedeutet, dass der Zeigefinger (Z), Mittelfinger (M) oder Ringfinger (R) der entsprechenden Hand gedrückt sein muss. 5
2. Fingersatz des Bleauregard Auswahl der entsprechenden Tonhöhe und Oktave basierend auf Gray-Codes nach Beauregard (1991). Der Zeigefinger (Z), Mittelfinger (M), Ringfinger (R) oder (K) kleiner Finger ist bei einem gefüllten Kreis gedrückt. 7
3. Vergleich der unterschiedlichen Bewegungen zur Tonhöhenauswahl 10
4. Mapping der Fingerkombinationen auf die Tonhöhe. Bei einer Eins ist der kleine Finger (K), Ringfinger (R), Mittelfinger (M) oder Zeigefinger (Z) gedrückt. . . . 24
5. Übersicht über die Versuchsbedingungen 43
6. Mittelwerte und Standardabweichungen der Reaktionszeiten in ms 44
7. Ergebnisse der t-Tests für abhängige Stichproben der mittleren Fehlerraten bei vier Noten, $*p < .017$ (Bonferroni-Korrektur) 47
8. Ergebnisse der t-Tests für abhängige Stichproben der mittleren Fehlerraten bei zwölf Noten, $*p < .017$ (Bonferroni-Korrektur) 47
9. Fragen zu dem freien Spielen und die entsprechenden arithmetischen Mittelwerte über alle Versuchsteilnehmer und die Standardabweichungen der Antworten . . . 50
10. Für das Instrument angepasste Aussagen der System Usability Scale, arithmetischer Mittelwert und Standardabweichung der Bewertungen 51

Literatur

Akai Professional, LP. (2015): „EWI USB/Docs & Downloads/Quickstart Manual [Rev.E].” URL http://www.akaipro.com/product/ewiusb. Zugriff 2015-07-26.

Anderson, J.R. (2013): *Cognitive Skills and Their Acquisition*. Carnegie Mellon Symposia on Cognition Series. Taylor & Francis.

Arfib, Daniel; Jean-Michel Couturier und Loïc Kessous (2004): „Design and use of some new digital musical instruments.” In: *Gesture-Based Communication in Human-Computer Interaction*. Springer, S. 509–518.

Backhaus, Claus (2009): *Usability-Engineering in der Medizintechnik: Grundlagen - Methoden - Beispiele*. VDI-Buch. Springer Berlin Heidelberg.

Baecker, R.M. und W.A.S. Buxton (1987): *Readings in human-computer interaction: a multidisciplinary approach*. Morgan Kaufmann readings series. M. Kaufmann.

Bangor, Aaron; Philip T. Kortum und James T. Miller (2008): „An empirical evaluation of the system usability scale.” In: *Intl. Journal of Human–Computer Interaction*, 24(6) S. 574–594.

Barbosa, Jerônimo et al. (2011): „Towards an evaluation methodology for digital music instruments considering performer's view: a case study.” In: *Proceedings of 13th Brazilian Symposium on Computer Music*. 13th Brazilian Symposium on Computer Music.

Beauregard, Gerald T. (1991): „Rethinking the design of wind controllers.” Masterarbeit. Hanover: Dartmouth College.

Bennett, J.L. (1984): *Visual display terminals: usability issues and health concerns*. Prentice-Hall.

Butler, Jennifer (2008): „Creating pedagogical etudes for interactive instruments.” In: *Proceedings of the International Conferences on New Interfaces for Musical Expression*. S. 77–80.

Cadoz, Claude (1988): „Instrumental gesture and musical composition.” In: *ICMC 1988-International Computer Music Conference*. S. 1–12.

Card, Stuart K.; William K. English und Betty J. Burr (1978): „Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT.” In: *Ergonomics*, 21(8) S. 601–613.

Danks, Mark (1997): „Real-time image and video processing in gem.” In: *Proceedings of the International Computer Music Conference*. S. 220–223.

Dix, A. (2004): *Human-computer Interaction*. Pearson/Prentice-Hall.

Doepfer Musikelektronik, GmbH (2015): „R2M Midi Ribbon Controller.” URL http://www.doepfer.de/R2M.htm. Zugriff 2015-21-12.

Fitts, Paul M. (1954): „The information capacity of the human motor system in controlling the amplitude of movement.” In: *Journal of experimental psychology*, 47(6) S. 381.

Gatzsche, Gabriel; Markus Mehnert und Christian Stöcklmeier (2008): „Interaction with tonal pitch spaces.” In: *8th International Conference on New Interfaces for Musical Expression NIME08*. S. 325.

Ghamsari, Mahtab; Amandine Pras und Marcelo M. Wanderley (2013): „Combining musical tasks and improvisation in evaluating novel digital musical instruments." In: *Proceedings of the 10th International Symposium on Computer Music Multidisciplinary Research, Marseille*. S. 506–515.

Godøy, Rolf Inge und Marc Leman (2010): *Musical Gestures: Sound, Movement, and Meaning*. Taylor & Francis.

Hick, William E. (1952): „On the rate of gain of information." In: *Quarterly Journal of Experimental Psychology*, 4(1) S. 11–26.

Holland, Simon; Katie Wilkie; Paul Mulholland und Allan Seago (2013a): *Music and Human-Computer Interaction*. Springer Series on Cultural Computing. Springer London.

Holland, Simon; Katie Wilkie; Paul Mulholland und Allan Seago (2013b): *Music and Human-Computer Interaction*. Springer Series on Cultural Computing. Springer London.

Hommel, Bernhard und Dieter Nattkemper (2011): *Handlungspsychologie. Planung und Kontrolle intentionalen Handelns*. Springer-Lehrbuch. Springer Berlin Heidelberg.

Hunt, Andy und Ross Kirk (1999): „Radical user interfaces for real-time control." In: *EUROMICRO Conference, 1999. Proceedings. 25th*, vol. 2. IEEE, S. 6–12.

Hunt, Andy und Ross Kirk (2000): „Mapping strategies for musical performance." In: *Trends in Gestural Control of Music*, 21 S. 231–258.

Hunt, Andy; Marcelo M. Wanderley und Ross Kirk (2000): „Towards a model for instrumental mapping in expert musical interaction." In: *Proceedings of the 2000 International Computer Music Conference*. S. 209–212.

Interlink Electronics, Inc. (2015): „FSR® 400 Series Data Sheet." URL http://www.interlinkelectronics.com/datasheets/Datasheet_FSR.pdf. Zugriff 2015-12-21.

Jordan, Patrick W. und Bernard Weerdmeester B. Thomas, Ian Lyall McClelland (1996): *Usability Evaluation In Industry*. Taylor & Francis.

Karnath, Hans-Otto und Peter Thier (2006): *Neuropsychologie*. Springer-Lehrbuch. Springer Berlin Heidelberg.

Kiefer, Chris; Nick Collins und Geraldine Fitzpatrick (????): „HCI methodology for evaluating musical controllers: A case study." In: *Proceedings of NIME*, vol. 8.

Lammers, Mark; Mark Kruger; Lela June Stoner; Richard Fuller und Debra Allyn (1996): „Biomechanics of musical performance: development of expertise in a natural task, playing the trombone." In: *Biomedical Engineering Conference, 1996., Proceedings of the 1996 Fifteenth Southern*. IEEE, S. 56–59.

Levitin, Daniel J.; Stephen McAdams und Robert L. Adams (2002): „Control parameters for musical instruments: a foundation for new mappings of gesture to sound." In: *Organised Sound*, 7(02) S. 171–189.

MacKenzie, I. Scott (1992): „Fitts' law as a research and design tool in human-computer interaction." In: *Human-computer interaction*, 7(1) S. 91–139.

Marshall, Mark T.; Max Hartshorn; Marcelo M. Wanderley und Daniel J. Levitin (2009): „Sensor choice for parameter modulations in digital musical instruments: Empirical evidence from pitch modulation." In: *Journal of New Music Research*, 38(3) S. 241–253.

Marshall, Mark T. und Marcelo M. Wanderley (2006): „Evaluation of sensors as input devices for computer music interfaces." In: *Computer Music Modeling and Retrieval*. Springer, S. 130–139.

Medeiros, Rodrigo; Filipe Calegario; Giordano Cabral und Geber Ramalho (2014): „Challenges in Designing New Interfaces for Musical Expression." In: *Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience*. Springer, S. 643–652.

Mey, Günter und Katja Mruck (2010): *Handbuch Qualitative Forschung in der Psychologie*. VS Verlag für Sozialwissenschaften.

Miranda, Eduardo Reck und Marcelo M. Wanderley (2006): *New Digital Musical Instruments: Control and Interaction Beyond the Keyboard*. Computer Music and Digital Audio Series. A-R Editions.

Moss, Pete und Bryan Cunitz (2005): „Haptic theremin: Developing a haptic musical controller using the Sensable Phantom Omni." In: *Proceedings of the International Computer Muisc Conference*. Barcelona, Spain, S. 275?277.

Müllensiefen, D.; B. Gingras; JJ L. Stewart und Musil (2012): „Goldsmiths musical sophistication index (gold-msi) v1."

O'modhrain, Sile (2011): „A framework for the evaluation of digital musical instruments." In: *Computer Music Journal*, 35(1) S. 28–42.

Paine, Garth (2013): „New Musical Instrument Design Considerations." In: *MultiMedia, IEEE*, 20(4) S. 76–84.

Palacio-Quintin, Cl'eo (2003): „The Hyper-Flute." In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Montreal, Canada, S. 206–207.

Pardue, Laurel und William Sebastian (2013): „Hand-Controller for Combined Tactile Control and Motion Tracking." In: *Proceedings of the International Conference on New Interfaces for Musical Expression*. Daejeon, Republic of Korea: Graduate School of Culture Technology, KAIST, S. 90–93.

Pew, Richard W. (1969): „The speed-accuracy operating characteristic." In: *Acta Psychologica*, 30 S. 16–26.

Puckette, Miller et al. (1996): „Pure Data: another integrated computer music environment." In: *Proceedings of the Second Intercollege Computer Music Concerts* S. 37–41.

Rovan, Joseph Butch; Marcelo M Wanderley; Shlomo Dubnov und Philippe Depalle (1997): „Instrumental gestural mapping strategies as expressivity determinants in computer music performance." In: *Kansei, The Technology of Emotion. Proceedings of the AIMI International Workshop*. Citeseer, S. 68–73.

Sawilowsky, Shlomo S und R Clifford Blair (1992): „A more realistic look at the robustness and Type II error properties of the t test to departures from population normality." In: *Psychological bulletin*, 111(2) S. 352.

Schubert, Patric (2013): „Die Anwendung nichtlinearer Verfahren zur charakterisierung der menschlichen Variabilität aus Zeitreihen." In: *Deutsche Zeitschrift für Sportmedizin*, 64(5) S. 132–140.

Shneiderman, Ben und Catherine Plaisant (2005): *Designing the User Interface: Strategies for Effective Human-computer Interaction*. Pearson/Addison Wesley.

Stoffregen, Paul und Robin Coon (2015): „Teensy USB Development Board." URL https://www.pjrc.com/teensy/index.html. Zugriff 2015-07-18.

Stowell, D.; A. Robertson; N. Bryan-Kinns und M. D. Plumbley (2009): „Evaluation of Live Human-computer Music-making: Quantitative and Qualitative Approaches." In: *Int. J. Hum.-Comput. Stud.*, 67(11) S. 960–975. doi:10.1016/j.ijhcs.2009.05.007.

Thon, Bernard und Claude Bonneviale (1995): „Performance on two-finger chords: Practice effects and advance information." In: *Human movement science*, 14(2) S. 247–273.

Troll Factory Rainer Habekost, e.K. (2015): „TFC Silikon Kautschuk Typ 13." URL http://www.trollfactory.de. Zugriff 2015-15-12.

Vertegaal, Roel und Barry Eaglestone (1996): „Comparison of input devices in an ISEE direct timbre manipulation task." In: *Interacting with Computers*, 8(1) S. 13–30.

Wanderley, Marcelo M.; Bradley W. Vines; Neil Middleton; Cory McKay und Wesley Hatch (2005): „The Musical Significance of Clarinetists' Ancillary Gestures: An Exploration of the Field." In: *Journal of New Music Research*, 34(1) S. 97–113.

Wanderley, Marcelo M.; Jean-Philippe Viollet; Fabrice Isart und Xavier Rodet (2000): „On the choice of transducer technologies for specific musical functions." In: *Proceedings of the 2000 International Computer Music Conference*. S. 244–247.

Wanderley, Marcelo M. und Nicolas M. Orio (2002): „Evaluation of Input Devices for Musical Expression: Borrowing Tools from HCI." In: *Computer Music Journal*, 26 S. 62–67.

Wetzlinger, Werner; Andreas Auinger und Michael Dörflinger (2014): „Comparing effectiveness, efficiency, ease of use, usability and user experience when using tablets and laptops." In: *Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience*. Springer, S. 402–412.

Winkler, Todd (1995): „Making motion musical: Gesture mapping strategies for interactive computer music." In: *ICMC Proceedings*. S. 261–264.

Wright, Matthew; Ryan J. Cassidy und Michael F. Zbyszyński (2004): „Audio and Gesture Latency Measurements on Linux and OSX." In: *Proceedings of the ICMC*. Miami, FL, USA, S. 423–429.

Yunik, M; M Borys und GW Swift (1985): „A digital flute." In: *Computer Music Journal* S. 49–52.

A. Versuchsanleitung und Fragebogen

Evaluation der Tonhöhensteuerung über Vier-Finger-Kombinationen

Einleitung

Vielen Dank, dass Sie sich dazu bereit erklären, an diesem Versuch teilzunehmen. Die Teilnahme ist freiwillig und Sie können den Versuch zu jedem Zeitpunkt abbrechen. Alle Daten werden anonym gespeichert und dienen rein wissenschaftlichen Zwecken. Der Versuch wird maximal eine Stunde dauern. Er besteht aus sechs kurzen Experimenten zur Tonhöhenauswahl mit einem neuen Controller und einem MIDI-Keyboard, freiem Spielen und einem Fragebogen. Falls Sie zwischendurch Fragen haben, können Sie sich jeder Zeit an den Versuchsleiter wenden.

Vorbereitung

Bitte lesen Sie die Versuchsanleitung aufmerksam durch und unterschreiben Sie die Einwilligungserklärung auf der letzten Seite, sofern Sie damit einverstanden sind.

Beschreibung des Controllers

Die vier Tasten an der Vorderseite dienen der Tonhöhenauswahl. Durch drücken von unterschiedlichen Kombinationen, können die 12 Halbtöne einer Oktave ausgewählt werden. Um die Fingerkombinationen jeweils einer Tonhöhe zuzuweisen wird das Prinzip der Binärzahlen angewendet. Die Halbtöne einer Oktave von C bis H, werden den Zahlen von 1 bis 12 zugewiesen. Werden diese als Binärcode mit vier Bit dargestellt und jede Stelle der Binärzahl einem Finger zugewiesen, ergibt sich die entsprechende Fingerkombination. Ausgehend von dem Grundton C, wenn nur der Zeigefinger gedrückt ist, kann die Tonhöhe in Halbtonschritten bis zum H, wenn kleiner Finger und Ringfinger gedrückt sind, ausgewählt werden. Eine Grifftabelle mit allen Fingerkombinationen ist im Anhang zu finden. Bei der Auswahl der Tonhöhe ist es wichtig, dass alle beteiligten Tasten möglichst gleichzeitig gedrückt werden, da ansonsten ungewollte Zwischentöne gespielt werden, ähnlich wie bei einer Flöte.

Auf der Rückseite des Controllers befinden sich drei Druckschalter zur Oktavauswahl, welche anfangs nicht benötigt werden.

Das Experiment wird für zwei unterschiedliche Spielmodi durchgeführt. Bei der ersten Variante wird der Ton gespielt sobald die Tasten zur Tonhöhenauswahl gedrückt wurden. Bei der zweiten Variante wird der Ton gespielt, wenn zusätzlich Druck auf die runde Fläche am Kopfende ausgeübt wird.

Versuchsanleitung Teil A: Reaktionszeiten

Bei der Messung der Reaktionszeit werden die zu spielenden Tonhöhen in zufälliger Reihenfolge als Grifftabelle auf dem Monitor vor Ihnen angezeigt. Die zum Spielen des Tones benötigten Tasten werden rot eingefärbt und die von Ihnen gedrückten Tasten in grün. Spielen Sie den Ton so schnell und fehlerfrei wie möglich, nachdem dieser angezeigt wurde. Nach Spielen der richtigen Tonhöhe, wird sofort die nächste angezeigt. Der Versuch wird in zufälliger Reihenfolge für sechs Versuchsbedingungen durchgeführt: Für die Klaviatur, den Controller mit zwei unterschiedlichen Spielmodi, jeweils mit vier und zwölf möglichen Tonhöhen.
Bei dem Controller ist jeder Finger fest einer Taste zugeordnet, gleiches gilt für den Versuch mit der Klaviatur mit vier möglichen Tonhöhen. Dabei werden nur die weißen Tasten des angezeigten Klaviaturausschnittes verwendet. Bei dem Versuch mit der Klaviatur und zwölf möglichen Tonhöhen, gibt es keine feste Zuordnung der Finger. Verwenden Sie für alle Versuche Ihre bevorzugte Hand.
Zur Eingewöhnung werden für jede Bedingung 24 Wiederholungen durchgeführt. Anschließend folgt der eigentlichen Versuch mit 60 Wiederholungen. Der Versuch beginnt wenn der Zähler auf dem Monitor von 3 bis 1 heruntergezählt hat. Füllen Sie nach jedem der sechs Versuche den kurzen Fragebogen aus.

Versuchsanleitung Teil B: Freies Spielen

Bei dem freien Spielen stehen alle Funktionen des Controllers zur Verfügung, die Oktave kann durch drücken auf die Taster auf der Rückseite ausgewählt werden und bleibt beibehalten, der Ton wird über den Druck auf die Fläche auf der Oberseite ausgelöst und moduliert. Spielen Sie für maximal 10 Minuten frei mit dem Controller und versuchen Sie den Funktionsumfang des Controllers auszuloten.

Versuchsanleitung Teil C: Fragebogen

Füllen Sie Abschließend den Fragebogen aus. Die angegeben Informationen werden ausschließlich zur Auswertung des Experimentes verwendet.

Einwilligungserklärung
Ich habe das Informationsblatt zu dem Experiment gelesen und verstanden. Ich hatte die Möglichkeit, Fragen zu stellen und verstand die Antworten. Ich wurde darüber informiert, dass ich jederzeit und ohne Angabe von Gründen diese Zustimmung widerrufen kann, ohne dass dadurch Nachteile für mich entstehen. Mir ist bekannt, dass meine Daten anonym gespeichert und ausschließlich für wissenschaftliche Zwecke verwendet werden. Hiermit willige ich ein, an dem Versuch teilzunehmen und erkläre mich damit einverstanden, dass die angegebenen Informationen zur Auswertung des Experimentes verwendet werden.

_____	_____	_____
Teilnehmer/in	Versuchsleiter/in	Ort, Datum

Fragen zu den Reaktionstest Nr. _____

1. Wie schwer fanden Sie die Tonhöhensteuerung?

   Sehr leicht ☐ ☐ ☐ ☐ ☐ Sehr schwer

2. Wie gut konnten Sie die visuellen Stimuli interpretieren?

   Sehr leicht ☐ ☐ ☐ ☐ ☐ Sehr schwer

3. Wie häufig war Ihr Blick auf die Tasten gerichtet?

   Nie ☐ ☐ ☐ ☐ ☐ Sehr häufig

4. Wie intuitiv empfanden Sie die Tonhöhenauswahl?

   Nicht intuitiv ☐ ☐ ☐ ☐ ☐ Sehr intuitiv

5. Wie zuverlässig schätzen Sie die Art der Tonhöhenauswahl ein?

   Unzuverlässig ☐ ☐ ☐ ☐ ☐ Sehr zuverlässig

6. Wie hoch schätzen Sie die benötigten motorischen Fähigkeiten ein?

   Gering ☐ ☐ ☐ ☐ ☐ Sehr hoch

Fragen zu den Reaktionstest Nr. _____

1. Wie schwer fanden Sie die Tonhöhensteuerung?

   Sehr leicht ☐ ☐ ☐ ☐ ☐ Sehr schwer

2. Wie gut konnten Sie die visuellen Stimuli interpretieren?

   Sehr leicht ☐ ☐ ☐ ☐ ☐ Sehr schwer

3. Wie häufig war Ihr Blick auf die Tasten gerichtet?

   Nie ☐ ☐ ☐ ☐ ☐ Sehr häufig

4. Wie intuitiv empfanden Sie die Tonhöhenauswahl?

   Nicht intuitiv ☐ ☐ ☐ ☐ ☐ Sehr intuitiv

5. Wie zuverlässig schätzen Sie die Art der Tonhöhenauswahl ein?

   Unzuverlässig ☐ ☐ ☐ ☐ ☐ Sehr zuverlässig

6. Wie hoch schätzen Sie die benötigten motorischen Fähigkeiten ein?

   Gering ☐ ☐ ☐ ☐ ☐ Sehr hoch

…

Fragen zu dem freien Spielen:

1. Wie aufregend empfanden Sie das Spielen des Instrumentes?
   Nicht aufregend ☐ ☐ ☐ ☐ ☐ Sehr aufregend
2. Wie inspirierend empfanden Sie das Spielen des Instrumentes?
   Nicht inspirierend ☐ ☐ ☐ ☐ ☐ Sehr inspirierend
3. Wie fließend empfanden Sie das Spielen des Instrumentes?
   Nicht fließend ☐ ☐ ☐ ☐ ☐ Sehr fließend
4. Wie vertraut empfanden Sie das Spielen des Instrumentes?
   Nicht vertraut ☐ ☐ ☐ ☐ ☐ Sehr vertraut
5. Wie direkt empfanden Sie die Interaktion?
   Nicht direkt ☐ ☐ ☐ ☐ ☐ Sehr direkt
6. Wie geeignet finden Sie das Instrument?
   Nicht geeignet ☐ ☐ ☐ ☐ ☐ Sehr geeignet
7. Wie schnell hat das Instrument auf Ihre Gesten reagiert?
   Nicht ☐ ☐ ☐ ☐ ☐ Sehr schnell
8. Wie genau hat das Instrument auf Ihre Gesten reagiert?
   Nicht ☐ ☐ ☐ ☐ ☐ Sehr genau
9. Als wie anwenderfreundlich empfanden Sie das Instrument?
   Nicht anwenderfreundlich ☐ ☐ ☐ ☐ ☐ Sehr anwenderfreundlich
10. Wie umfangreich schätzen Sie die musikalischen Möglichkeiten des Instrumentes ein?
    Nicht umfangreich ☐ ☐ ☐ ☐ ☐ Sehr umfangreich
11. Wie schnell haben Sie Fortschritte beim Erlernen gemacht?
    Sehr langsam ☐ ☐ ☐ ☐ ☐ Sehr schnell
12. Wenn Sie die Möglichkeit hätten, würden Sie das Instrument in Zukunft noch einmal verwenden?
    Nein ☐ ☐ ☐ ☐ ☐ Sehr gerne
13. Wie gut gefällt Ihnen der Klang des Instrumentes?
    Gefällt mir absolut nicht ☐ ☐ ☐ ☐ ☐ Gefällt mir sehr gut
14. Haben Sie weiteres Feedback zu dem Instrument? Was hat Ihnen daran gefallen oder nicht gefallen? Haben Sie Verbesserungsvorschläge? (freier Text)

Fragebogen zur Usability des Instrumentes

Bitte umkreisen Sie die am besten zutreffende Kategorie:	1 Stimme ganz & gar nicht zu	2 Stimme nicht zu	3 Weder noch	4 Stimme zu	5 Stimme voll und ganz zu
Ich denke, ich würde das Instrument gerne weiterhin spielen.	1	2	3	4	5
Ich finde das Instrument unnötig kompliziert.	1	2	3	4	5
Ich denke das Instrument war einfach anzuwenden.	1	2	3	4	5
Ich würde die Unterstützung eines Lehrers benötigen um damit Spielen zu lernen.	1	2	3	4	5
Die Eingabemöglichkeiten sind gut in das Instrument integriert.	1	2	3	4	5
Es gab zu viele Unstimmigkeiten bei der Anwendung.	1	2	3	4	5
Ich könnte mir vorstellen, dass die meisten Menschen das Instrument schnell erlernen können.	1	2	3	4	5
Ich finde, die Bedienung des Instrumentes war umständlich.	1	2	3	4	5
Ich fühle mich mit der Anwendung des Instrumentes sehr vertraut.	1	2	3	4	5
Ich muss noch eine Menge lernen bis ich mit dem Spielen des Instrumentes fortfahren kann.	1	2	3	4	5
Es ist bei der Erprobung des Instrumentes zu Problemen bzw. Fehlbedienungen gekommen? ☐ nein ☐ ja, es sind folgende Probleme aufgetreten:					
Ich halte das Instrument insgesamt für geeignet weil:					

Bitte füllen Sie aus oder umkreisen Sie die am meisten zutreffende Kategorie:

Ich würde mich selbst nicht als Musiker/in bezeichnen.
Stimme ganz und gar nicht zu ☐ ☐ ☐ ☐ ☐ ☐ ☐ Stimme voll und ganz zu

Ich bin noch nie für meine musikalischen Fähigkeiten gelobt worden.
Stimme ganz und gar nicht zu ☐ ☐ ☐ ☐ ☐ ☐ ☐ Stimme voll und ganz zu

Ich habe **0 / 1 / 2 / 3 / 4-6 / 7-10 / 11 oder mehr** Live-Events als Zuschauer innerhalb der letzten 12 Monate besucht.

Ich habe **0 / 0,5 / 1 / 2 / 3 / 4-6 / 7 oder mehr** Jahre Unterricht in Musiktheorie (außerhalb der Schule) erhalten.

Ich habe **0 / 1 / 2 / 3 / 4-5 / 6-9 / 10 oder mehr** Jahre Musikunterricht auf einem Instrument in meinem bisherigen Leben gehabt.

Ich höre jeden Tag aufmerksam Musik für **0-15min / 15-30min / 30-60min / 60-90min/ 2 Std / 2-3 Std / 4 Std oder mehr**.

Das Instrument, welches ich am besten spiele ist _____

Ich habe regelmäßig und täglich ein Instrument für **0 / 1 / 2 / 3 / 4-5 / 6-9 / 10 oder mehr** Jahre geübt.

An dem Höhepunkt meines Interesses habe ich mein Hauptinstrument **0 / 0,5 / 1 / 1,5 / 2 / 3-4 / 5 oder mehr** Stunden pro Tag geübt.

Ich kann **0 / 1 / 2 / 3 / 4 / 5 / 6 oder mehr** verschiedene Instrumente spielen.

Wenn Sie ein weiteres Instrument mit Klaviatur gespielt haben:

Ich habe regelmäßig und täglich ein Instrument mit Klaviatur für **0 / 1 / 2 / 3 / 4-5 / 6-9 / 10 oder mehr** Jahre geübt.

An dem Höhepunkt meines Interesses habe ich das Instrument mit Klaviatur **0 / 0,5 / 1 / 1,5 / 2 / 3-4 / 5 oder mehr** Stunden pro Tag geübt.

Wenn Sie ein weiteres Instrument mit Ventiltasten gespielt haben:

Ich habe regelmäßig und täglich ein Instrument mit Ventiltasten für **0 / 1 / 2 / 3 / 4-5 / 6-9 / 10 oder mehr** Jahre geübt.

An dem Höhepunkt meines Interesses habe ich das Instrument mit Ventiltasten **0 / 0,5 / 1 / 1,5 / 2 / 3-4 / 5 oder mehr** Stunden pro Tag geübt.

Ihr **Alter**: _____ Jahre

Geschlecht: weiblich / männlich

Aktuelle Haupttätigkeit: _____

Grifftabelle

Finger												
Z	●	○	●	○	●	○	●	○	●	○	●	○
M	○	●	●	○	○	●	●	○	○	●	●	○
R	○	○	○	●	●	●	●	○	○	○	○	●
K	○	○	○	○	○	○	○	●	●	●	●	●
Note	**C**	**C#**	**D**	**D#**	**E**	**F**	**F#**	**G**	**G#**	**A**	**A#**	**H**
Nr.	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**

Mapping der Fingerkombinationen auf die Tonhöhen für den Zeigefinger (Z), Mittelfinger (M), Ringfinger (R) und kleinen Finger (K).

B. Verwendete Materialien

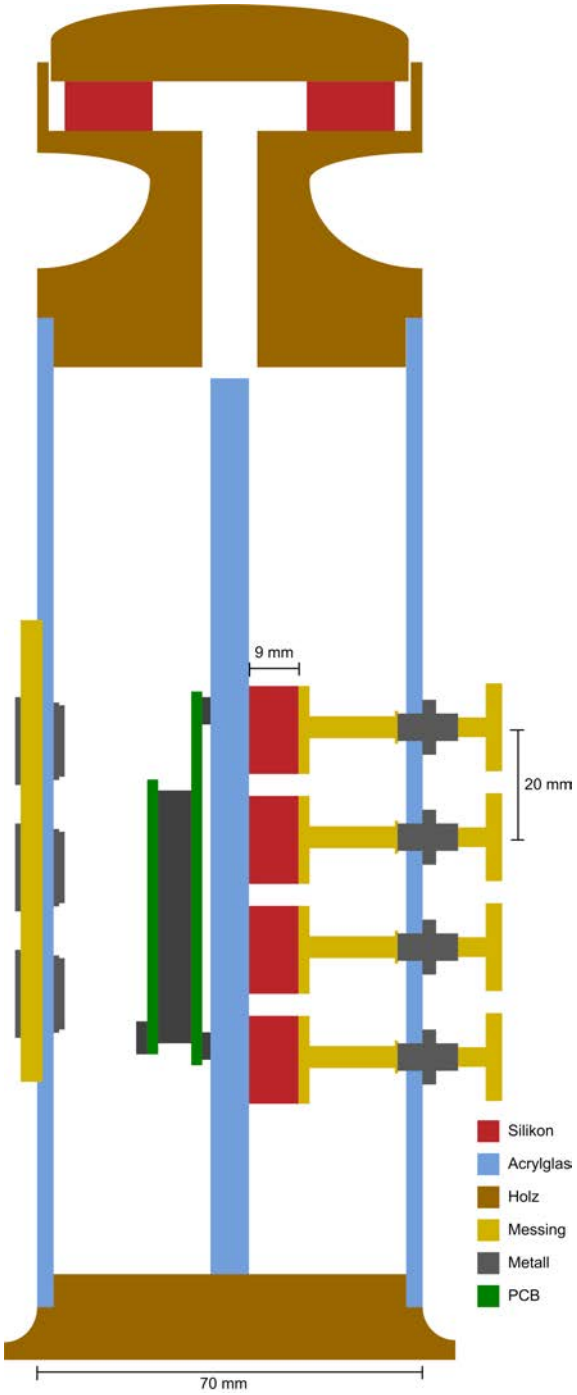

Abbildung 20: Übersicht über verwendete Materialien, dargestellt im Querschnitt des Controllers

C. Bilder

Abbildung 21: Seitenansicht des Controllers

Abbildung 22: Ansicht der Rückseite des Controllers

D. Programmablaufplan

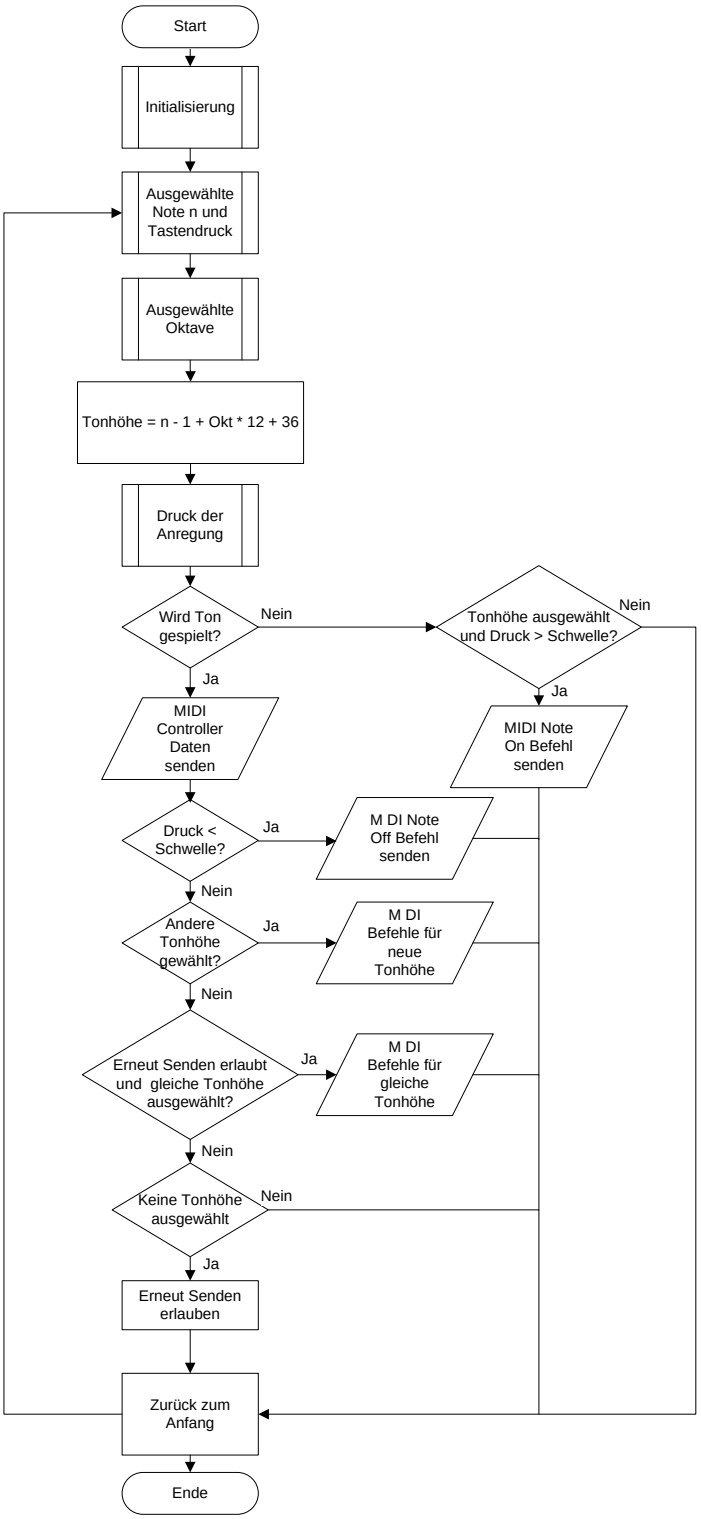

Abbildung 23: Programmablaufplan der Firmware mit MIDI Implementierung

E. Programmcode

```
1:  /*
2:  Filename:  Controller_03.ino
3:  Version:   0.3
4:
5:  Project:   Master Thesis (Controller development for pitch controll via finger combinations)
6:
7:  Usage  :   Arduino sketch for Teensy 3.1 board with 8 FSR sensors between +5V to ADC and
8:             10 kOhm resistor between ADC and GND. Sensors connected to A0-A3 are used to select
9:             the note by (finger) combinations of the 4 sensors (pitch). MIDI controller
10:            information are driven from the 4 sensors conncted to A6-A9 (modulation).
11:       Sum is used to controll velocity and #10.
12:       3 piezo scitches are connected between digital pins 2-4 and GND,
13:            using internal pull-up resistors.
14:            Note on/off information are send corresponding to the selected key
15:            combination and octave, when the top plate is pushed. Keeping the top plate
16:       pushed and releaseing the keys continues
17:            playing the note.
18:            Keeping latest selected octave until new octave is selected.
19:
20:  Author :  Gabriel Treindl
21:  Date   :  06.06.15
22:  Updated:  <11 Feb 2016 15:22, mh>
23: */
24:
25: // Treshold to detect key pressed
26: #define NOTETHR 200
27: // Treshold for to detect controller FSR
28: #define CTRLTHR 20
29: // Lowest MIDI note
30: #define BASENOTE 36
31:
32: // Initialite counters, to send MIDI controller data after 200 code cycles
33: int sendCounter = 0;
34: int controlCount = 0;
35: // Store analog FSR values
36: int analogPresFSR[4];
37: int analogPresExpFSR[4];
38: int analogKeyPins[4] = {3, 2, 1, 0};
39: int analogPresPins[4] = {9, 6, 8, 7};
40:
41: // Smoothing parameter for modulation
42: float analogPresSmooth[6];
43: // Store values for recursive filter
44: float analogPresSmooth_1[6] = {0.0, 0.0, 0.0, 0.0, 0.0, 0.0};
45: float analogPresSmoothCtrl_1[4] = {0.0, 0.0, 0.0, 0.0};
46: // Filter smoothing parameter
47: float alpha = 0.93;
48:
49: // Pins of piezo buttons to select octave
50: int digitalPins[4] = {4, 3, 2, 5};
51: // Number of octave buttons
52: int numButtons = 3;
53:
54: // Store analog key values (pitch)
55: int analogKeyFSR[4];
56: // Store selected note
```

```
57: int binaryNote_1 = 0;
58: // Binary note to be send (after time window)
59: int binaryNoteSend;
60: // Counter to verify selected note (time window = 150 cycles, about 46 ms)
61: int counterNote = 0;
62: // Note playing status (monophonic)
63: byte noteIsOn = 0;
64: // Threshold to detect push
65: int upThr  = 200;
66: int dwnThr = 50;
67: // Sum of pressure
68: int sumExp = 0;
69: // Store sum value for next cycle
70: int sumLinear = 0;
71: int sumLinear_1 = 0;
72:
73: // MIDI Data
74: // Controller number
75: int controller[8] = {4, 5, 6, 7, 80, 81, 10, 2};
76: // Controller value
77: int contrValue[8] = {0, 0, 0, 0, 0, 0, 0, 0};
78: // Previous controller value
79: int contrValue_1[8] = {0, 0, 0, 0, 0, 0, 0, 0};
80: // Velocity
81: int velocity;
82: // Selected octave
83: int octave = 0;
84: // last note
85: int note_1 = 0;
86: int channel = 1;
87: // Allow to send same note again (binary)
88: int sendAgain = 0;
89:
90: void setup()
91: {
92:   // Octave Buttons initialization
93:   for(int n = 0; n < numButtons; n++){
94:     // Digital iputs
95:     pinMode(digitalPins[n], INPUT);
96:     // Internal pullup
97:     digitalWrite(digitalPins[n], HIGH);
98:   }
99: }
100:
101: void loop()
102: {
103:   /* Pitch */
104:   int binaryNote = 0;
105:
106:   // Initialize average pressure of selected keys
107:   int meanKeyPressure = 0;
108:
109:   // Get binary note and average key pressure
110:   getNote(&binaryNote, &meanKeyPressure);
111:
112:   // Wait 150 cycles if note has changed
113:   // Not changed
114:   if (binaryNote == binaryNote_1)
```

```
115:     binaryNoteSend = binaryNote;
116:   // Time out or no note selected
117:   else if (counterNote == 150 || binaryNote == 0)  {
118:     binaryNoteSend = binaryNote;
119:     binaryNote_1 = binaryNote;
120:     counterNote = 0;
121:   }
122:   // Note has changed
123:   else {
124:     binaryNoteSend = binaryNote_1;
125:     counterNote++;
126:   }
127:
128:   /* Octave */
129:   // Routine to check octave buttons
130:   getOctave(&octave);
131:
132:   /* Amplitude */
133:   // Exponential mapped pressure for
134:   sumExp = 0;
135:   // Einear mapped pressure for first note on
136:   sumLinear = 0;
137:   // Value for data smoothing
138:   float analogSumSm;
139:
140:   for(int k = 0; k<4; k++){
141:     // Read all analog values
142:     int analogSum = analogRead(analogPresPins[k]);
143:
144:     // Sum of all 4 analog values (linear)
145:     sumLinear += analogSum;
146:
147:     // Exponential mapping of pressure
148:     analogSumSm = 0.35 * (exp(float(analogSum)/128.0)-1.0);
149:
150:     // Smoothing values of each sensor (float number)
151:     if(analogSumSm > analogPresSmooth_1[k])
152:       // Faster attack
153:       alpha = 0.90;
154:     else
155:       // Slower release
156:       alpha = 0.95;
157:
158:     // Recursive filter
159:     analogSumSm = (1.0-alpha) * analogSumSm + alpha * analogPresSmooth_1[k];
160:     analogPresSmooth_1[k] = analogSumSm;
161:
162:     // Current sum of all exponetial mapped sensor data
163:     sumExp += int(analogSumSm);
164:   }
165:   // Average value
166:   sumExp /= 4;
167:   sumLinear /=4;
168:
169:   /* Send Note */
170:   // Linear mapping for velocity between 1 and 127
171:   velocity = map(sumExp,0,1023,0,127);
172:   // Make sure it's not out or range
```

```
173:   if(velocity > 127)
174:     velocity = 127;
175:   else if(velocity < 1)
176:     velocity = 1;
177:
178:   // Caltulate MIDI note number
179:   int note = binaryNoteSend + 12*octave + BASENOTE - 1;
180:
181:   // Check for sending MIDI note on/off message
182:   if(noteIsOn == 0) //note is off, not playing
183:     {
184:       if(sumLinear > upThr && binaryNoteSend > 0)
185:       {
186:           // velocity
187:           int vel = velocity;
188:           // difference to previous value
189:           int diff = 10;
190:           //first note on
191:           if(binaryNote_1 != 0){
192:             int rise = sumLinear;
193:             // Wait for first maximum
194:             while (diff > 0){
195:                 for(int k = 0; k<4; k++)
196:                   rise += analogRead(analogPresPins[k]);
197:                 // average pressure
198:                 rise /= 4;
199:                 // Difference to previouis pressure
200:                 diff = rise - sumLinear;
201:                 // current pressure
202:                 sumLinear = rise;
203:              }
204:              // Map value to MIDI range
205:              vel = map(sumLinear,upThr,1023,1,127);
206:              if(vel > 127)
207:                vel = 127;
208:              else if(vel < 1)
209:                vel = 1;
210:          }
211:          // Send MIDI note and update status
212:          usbMIDI.sendNoteOn(note, vel, channel);
213:          noteIsOn = 1;
214:          sendAgain = 0;
215:          // Remember last send note
216:          note_1 = note;
217:       }
218:    }
219:    else //note is on, playing
220:    {
221:       // Send after 25 cycles
222:       controlCount++;
223:       if(controlCount == 25){
224:         // Send xy-controller data
225:         sendController();
226:         controlCount = 0;
227:       }
228:       // Pressure below threshold -> Note Off
229:       if(sumLinear < 80 )
230:       {
```

```
231:         usbMIDI.sendNoteOff(note_1, 0, channel);
232:         noteIsOn = 0;
233:        }
234:        // Other note selected -> Send note On & note Off
235:        else if(note != note_1 && binaryNoteSend > 0)
236:        {
237:          // Send note on for new and note off for last note
238:          usbMIDI.sendNoteOn(note, velocity, channel);
239:          usbMIDI.sendNoteOff(note_1, 0, channel);
240:          note_1 = note;
241:          sendAgain = 0;
242:        }
243:        // Same note selected again -> Send note Off & note On
244:        else if(sendAgain && binaryNoteSend > 0){
245:          usbMIDI.sendNoteOff(note, 0, channel);
246:          usbMIDI.sendNoteOn(note, velocity, channel);
247:          sendAgain = 0;
248:         }
249:         // allow to send same note again
250:         else if(binaryNoteSend == 0)
251:           sendAgain = 1;
252:   }
253:
254:   /* Send Controller every 150 cycles*/
255:   sendCounter++;
256:   if (sendCounter==150){
257:     // Send key pressure
258:     usbMIDI.sendControlChange(controller[6], meanKeyPressure, channel);
259:     // MIDI Controller data
260:     usbMIDI.sendControlChange(controller[7], velocity , channel);
261:     // Reset counter
262:     sendCounter =  0;
263:   }
264: }
265:
266: /* getNote: Reading analog values and calculate selected binary note and key pressure
267:    Input parameters: Pointer to binaryNote
268:                      Pointer to meanKeyPressure
269:    Output parameters: - (editing in place)
270:  */
271: void getNote(int* binaryNote, int* meanKeyPressure)
272: {
273:   // Number of selected keys
274:   int numKey = 0;
275:   // Loop over all keys
276:   for(int k = 0; k<4; k++)
277:   {
278:     // Read analog values
279:     analogKeyFSR[k] = analogRead(analogKeyPins[k]);
280:
281:     // Set Bites according to pressed keys
282:     if(analogKeyFSR[k] > NOTETHR)
283:     {
284:       bitWrite(*binaryNote, 3-k, 1);
285:       *meanKeyPressure += analogKeyFSR[k]-NOTETHR;
286:       numKey++;
287:     }
288:   }
```

```
289:   // Average pressure by dividing sum by number of pressed keys
290:   if (numKey > 0)
291:     *meanKeyPressure = *meanKeyPressure/numKey;
292:   else
293:     *meanKeyPressure = 0;
294:   // Map to MIDI range
295:   *meanKeyPressure = map(*meanKeyPressure, 0, 1023-NOTETHR, 0, 127);
296:   // Limit to 7 bit
297:   if(*meanKeyPressure > 127)
298:     *meanKeyPressure = 127;
299:   else if(*meanKeyPressure < 1)
300:     *meanKeyPressure = 0;
301: }
302:
303: /* Octave */
304: void getOctave(int* octave)
305: {
306:   // Check status of all buttons
307:   for(int n = 0; n < numButtons; n++){
308:     // if pressed
309:     if (digitalRead(digitalPins[n]) == LOW)
310:       *octave = n;
311:   }
312: }
313:
314: /* MIDI Controller Data */
315: // Modulation
316: void sendController(void)
317: {
318:   // Initialize data
319:   int ctrlVal[4];
320:   int analogIn;
321:   float analogPresSm;
322:   // Loop over 4 ADC inputs
323:   for(int k = 0; k<4; k++)
324:   {
325:       // Read value from ADC
326:       analogIn = analogRead(analogPresPins[k]);
327:       // Convert to float
328:       analogPresSm = float(analogIn);
329:
330:       // Smoothing values of each sensor (float number)
331:       if(analogPresSm > analogPresSmoothCtrl_1[k])
332:         // Faster attack
333:         alpha = 0.85;
334:       else
335:         // Slower release
336:         alpha = 0.90;
337:
338:       // Recursive filter
339:       analogPresSm = (1.0-alpha) * analogPresSm + alpha * analogPresSmoothCtrl_1[k];
340:       analogPresSmoothCtrl_1[k] = analogPresSm;
341:       // Convert to intiger
342:       ctrlVal[k] = int(analogPresSm);
343:      // Limit to MIDI data range
344:      int ctrlMIDI = map(ctrlVal[k], 0, 1023, 0, 127);
345:       // Make sure within limits
346:       if(ctrlMIDI > 127)
```

```
347:         ctrlMIDI = 127;
348:       else if(ctrlMIDI < 0)
349:         ctrlMIDI = 0;
350:      // Send controller data
351:      usbMIDI.sendControlChange(controller[k], ctrlMIDI, channel);
352:   }
353:   // Map "pressure" to absolute x-direction data
354:   int xdir = map(ctrlVal[2] - ctrlVal[0], -1023, 1023, 0, 127);
355:   // Make sure within limits
356:   if(xdir > 127)
357:     xdir = 127;
358:   else if(xdir < 0)
359:     xdir = 0;
360:
361:   // Send MIDI controller data if changed
362:   if(xdir != contrValue_1[4])
363:   {
364:     usbMIDI.sendControlChange(controller[4], xdir, channel);
365:     contrValue_1[4] = xdir;
366:   }
367:   // Map "pressure" to absolute y-direction data
368:   int ydir = map(ctrlVal[3] - ctrlVal[1], -1023, 1023, 0, 127);
369:   if(ydir > 127)
370:     ydir = 127;
371:   else if(ydir < 0)
372:     ydir = 0;
373:   // Send MIDI-controller data
374:   if(ydir != contrValue_1[5])
375:   {
376:     usbMIDI.sendControlChange(controller[5], ydir, channel);
377:     contrValue_1[5] = ydir;
378:   }
379: }
```

Quellcode 6: Programmcode des Teensy 3.1, USB Type = MIDI