DIPLOMARBEIT

Modellierung von EOG Daten für Übertragung und Verarbeitung auf mobilen Geräte

Eingereicht an der Technischen Universität Wien
Fakultät für Elektrotechnik und Informationstechnik,

unter der Leitung von

Ao.Uni.Prof. Dipl.-Ing. Dr.techn. Felix Breitenecker
Institute Number: 101
Institute of Analysis and Scientific Computing

by

Wolfgang Wild
Matr.Nr. 0125040

02.06.2020

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

TU Bibliothek
Your knowledge hub
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Erklärung zur Verfassung der Arbeit

Wolfgang Wild

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

_____ _____

(Ort, Datum) (Unterschrift Wolfgang Wild)

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

TU Bibliothek
Your knowledge hub
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Abstract

In the digitalised world, alternative input systems for people with disabilities and in areas where classical input systems are not possible for process-technical reasons are in demand. The focus of research and development is, among other things, on human movements, such as head movements or eye movements. These movements can be recorded by video analysis or by measurements of corresponding bio-signals. One of these bio-signals is caused by a potential shift on the retina in the eye, triggered by muscle movements. It can be measured by electrooculography (EOG). The task is on the one hand to bring this bio-signal to the processing device by a most robust signal acquisition or transmission and on the other hand to evaluate this signal. With the help of a model that describes the eye movements, an algorithm is developed that interprets the measurement signal and generates control commands.

This diploma thesis deals with the acquisition and evaluation of the EOG signal. It is developed in cooperation with the Austrian Institute of Technology, which is involved in two research projects of the Austrian Research Promotion Agency FFG.

The first part of the thesis describes the required basics and the techniques used. The second part deals with the model used. It is based on the analogy between the potential shift on the retina, caused by eye movements, to an electric dipole. In the third part of the thesis, the firmware of the provided bio-signal detector is extended in such a way that the signal acquisition can be improved by simple adjustment of the measurement parameters. By the created Android application, it can be guaranteed that the signal quality remains verifiable before and during the recording. The measured data then serve as a database for the development and application of the dipole model for eye movements. It is mapped in an algorithm, which should recognize predefined events from the eye movements. As output of the model, control signals for other applications are provided.

In the last part of the work a verification of the algorithm is performed, and the results are interpreted.

It becomes obvious that the dipole model can be used for eye movements, but the quality of the signal and the associated continuous calibration of the system still have to be worked on.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

TU Bibliothek
Your knowledge hub
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Kurzfassung

In der digitalisierten Welt sind alternative Eingabesysteme für Menschen mit Behinderung und in Bereichen, in denen klassische Eingabesysteme aus prozesstechnischen Gründen nicht möglich sind, gefragt. Im Fokus von Forschung und Entwicklung stehen unter anderem Bewegungen des Menschen, wie Kopfbewegungen oder Augenbewegungen. Diese Bewegungen können durch Videoanalyse erfasst werden, oder aber durch Messungen von korrespondierenden Biosignalen. Eines dieser Biosignale entsteht durch eine Potentialverschiebung auf der Netzhaut im Auge, ausgelöst durch Muskelbewegungen. Es kann mithilfe der Elektrookulographie (EOG) gemessen werden. Die Aufgabe besteht einerseits darin, dieses Biosignal durch eine möglichst robuste Signalerfassung bzw. -übertragung zum verarbeitenden Gerät zu bringen und zum anderen dieses Signal auszuwerten. Mithilfe von einem Modell, welches die Augenbewegungen beschreibt, wird ein Algorithmus, der das Messignal interpretiert und daraus Steuerbefehle generiert, entwickelt.

Diese Diplomarbeit beschäftigt sich mit der Erfassung und Auswertung des EOG Signals. Sie entsteht in Zusammenarbeit mit dem Austrian Institute of Technologie, welches sich im Rahmen zweier Forschungsprojekte der österreichischen Forschungsförderungsgesellschaft FFG mit diesem Thema befasst.

Im ersten Teil der Arbeit werden die benötigten Grundlagen und die verwenden Techniken beschrieben. Der zweite Teil befasst sich mit dem verwendeten Modell. Es beruht auf der Analogie zwischen der Potentialverschiebung auf der Netzhaut, ausgelöst durch Augenbewegungen, zu einem elektrischen Dipol. Im dritten Teil der Arbeit wird die Firmware des bereitgestellten Biosignalerfassers so erweitert, dass die Signalerfassung durch einfache Anpassung der Messparameter verbessert werden kann. Durch die erstellte Android Applikation kann gewährleistet werden, dass die Signalqualität vor und während der Aufzeichnung überprüfbar bleibt. Die gemessenen Daten dienen dann als Datenbasis für die Entwicklung und Anwendung des Dipol-Modelles für Augenbewegungen. Es wird in einem Algorithmus, welcher aus den Augenbewegungen vordefinierte Events erkennen soll, abgebildet. Als Ausgang des Modelles werden Steuersignale für andere Anwendungen bereitgestellt.

Im letzten Teil der Arbeit wird eine Verifikation des Algorithmus durchgeführt, und die Ergebnisse werden interpretiert.

Aus der Arbeit wird ersichtlich, dass das Dipol-Modell für Augenbewegungen verwendet werden kann, an der Qualität des Signals, bzw. der damit verbunden kontinuierlichen Kalibrierung des Systems, muss aber noch gearbeitet werden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

TU Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Danksagung

Begegnet uns jemand,
der uns Dank schuldig ist,
gleich fällt es uns ein.
Wie oft können wir jemand begegnen,
dem wir Dank schuldig sind,
ohne daran zu denken!

Johann Wolfgang von Goethe, Ottiliens Tagebuch, 1809

In diesem Sinne möchte ich mich bei meinen Eltern Elisabeth und Rudolf Wild, die mich immer unterstützt haben, sei es mit Rat und Tat oder auch nur finanziell, bedanken. Selbst wenn ich auf den falschen Weg geraten bin. Auch bei meiner Schwester Ursula, die immer zur Stelle war und mithalf die Arbeit Korrektur zu lesen.
Ein besonderer Dank gilt auch meinem Betreuer Professor Felix Breitenecker der mir einen tiefen Einblick in die Welt der Wissenshaft ermöglichte. Auch meinen Kollegen allen voran Andreas Oberleitner und Martin Bachler danke ich für ihre Unterstützung während der ganzen Zeit, die ich für diese Diplomarbeit gebraucht habe. Auch an vielen anderen Unterstützern werde ich nie vorbei gehen ohne daran zu denke, welchen Dank ich ihnen schuldig bin, für ihren Beitrag zum erfolgreichen Abschluss dieser Arbeit.

Der letzte und größte Danke gebührt meiner Lebensgefährtin Frau Christine Kaltenbrunner, die mich in schwierigen Zeiten aufgefangen hat und mich motivierte diese Arbeit fertig zu stellen. Ob bei Tag oder in der Nacht, immer hat sie ein offenes Ohr für mich oder ein offenes Auge um diese Arbeit Korrektur zu lesen. Die es versteht mich zu beraten, ohne Kritik an mir zu üben.

Danke

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Inhaltsverzeichnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Abkürzungsverzeichnis

A/D	Analog-Digital-Umsetzer
ADT	Android Development Tools
AIT	Austrian Intitute of Technologie
API	Application Programming Interface
CTS	Clear to Send
DLL	Dynamic Link Library
EEG	Elektroenzephalografie
EKG	Elektrokardiogramm
EOG	Elektrookulografie
ERG	Elektroretinographie
FFG	Österreischische Forschungsfördergesellschaft
HAL	Hardware Abstraction Layer
HID	Human Interface Deices
IDE	Integrated development environment
IC	Integrated Circuit
L2CAP	Logical Link Control and Adaption Protocol
LMP	Link Manager Protocol
OSI	Open Systems Interconnection Model
PC	Personal Computer
PCM	Puls Code Modulation
RFCOMM	Radio Frequency Communication
RTS	Request to Send
SIG	Bluetooth Special Interest Group
SDK	Software Development Kit
SDP	Service Discovery Protocol
SPP	Serial-Port-Profil
TI	Texas Instruments
UART	Universal Asynchonour Receiver Transmitter
UUID	Universally Unique Identifier

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

1. Einleitung

1.1 Motivation

Personal Computer (PC) sind aus unserem Alltag kaum noch wegzudenken, weder aus der Arbeit noch aus der Freizeit. Auch viele andere technischen Systeme sind mittlerweile Computer gesteuert, sei es direkt am Gerät oder über vernetzte Systeme. So können viele Geräte mit dem Smartphone oder einem Tablet verwaltet werden. Die Verwendung und Steuermöglichkeiten dieser mobilen Geräte nimmt ständig zu. Die benötigten Eingaben werden, durch ihren Benutzer, mit Hilfe eines sogenannten Human Interface Devices (HID) gemacht. Die häufigsten Eingabesysteme sind Maus oder Tastatur, beide HIDs haben gemeinsam, dass sie von Hand zu steuern sind. Dies ist jedoch nicht immer möglich, entweder aus medizinischen Gründen, wie bei einer Behinderung, oder aus situationsbedingten Gründen wie z.B.: wenn keine Hand frei ist. Um in diesen Fällen auch eine Eingabe zu ermöglichen, gibt es unterschiedliche Ansätze wie Spracherkennung, Pupillen- und Kopfbewegungen oder Gestensteuerung. Ein großer Nachteil an solchen Systemen ist die Ortsgebundenheit. Sie wird häufig durch fix installierte Kameras oder einen nicht änderbaren Arbeitsbereich vorgegeben. Aber auch Akzeptanzprobleme von der Umgebung, wie es bei Sprachsteuerung oder "auffälligen" Handlungen vorkommt, können zum nicht Benutzen solcher Systeme führen. Immer aktueller werden auch Bedenken hinsichtlich der Privatsphäre und der Weiterverarbeitung der gesammelten Daten, die optische oder akustische Eingabesysteme produzieren.
Diese Arbeit beschäftigt sich mit dem Lösungsansatz: elektrische Biosignale zu erfassen und daraus Eingaben für computergestützte Systeme zu erstellen. Biosignale sind vor allem aus der Diagnostik in der Medizin bekannt wie z.B.: Elektrokardiogramm (EKG) oder auch Elektroenzephalographie (EEG). Das Erfassen von Biosignalen erfolgt für alle Arten gleich und kann daher auch für jedes beliebige elektrische Biosignal angewendet werden.
In dieser Arbeit wird das Hauptaugenmerk auf die Elektrookulographie (EOG), d.h. auf Biopotentiale, die abhängig von Augenbewegungen sind, gelegt. Der Vorteil zu anderen existierenden System, die auf die Bewegung der Augen reagieren, ist die Flexibilität der Datenerfassung, da keine Kamera genau auf die Augen gerichtet sein muss. Auch die Bedenken hinsichtlich der Privatsphäre entfallen durch die Verwendung eines Biosignales. Damit aus einem Biosignal Steuerbefehle für ein computerbasierendes System errechnet werden können, müssen die Signale zuverlässig erfasst und auf das zu verarbeitende Gerät übertragen werden. Damit die Anwendung möglichst mobil ist, erfolgt die Datenübertragung kabellos. Auf dem Gerät sorgt ein Algorithmus dafür, dass aus den Messerwerten Steuerbefehle generiert werden. Für das Entwickeln von Algorithmen werden echte Daten, die vorher aufgezeichnet werden, benötigt. Um die Qualität der gemessenen Daten zu verbessern und um Fehler vor der Aufzeichnung frühzeitig erkennen zu können, sollten die Daten vor und während der Messung überprüft werden. Meist lassen sich die Signale ganz einfach durch eine optische Überprüfung anhand eines zeitlichen Verlaufs überprüfen. Falls das Signal nicht den erwartenden Verlauf aufweist, kann durch Neupositionieren der Elektroden oder durch verändern der Messparameter die Signalqualität verbessert werden. Um Biosignale in möglichst verschieden Situation erfassen zu können, sollte das

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Messgerät möglichst klein sein, damit es den Träger so wenig wie möglich behindert. Die Geräte lassen sich sehr klein halten, wenn die Anzeige und Steuermöglichkeiten wie Knöpfe oder Schieberegler auf einem anderen vernetzten Gerät genutzt werden können. Um kein eigenes Anzeigegerät entwickeln zu müssen, können mobile Geräte wie Laptop, Tablet oder Smartphone verwendet werden. Damit ein effizienter Datenaustausch zwischen Geräten möglich ist, muss die Bandbreite der Verbindung ausreichend sein, sowie ein Kommunikationsprotokoll modelliert und eine Applikation für das Anzeigegerät programmiert werden. Die Auswertung erfolgt in einem ersten Schritt an einem PC. Sobald ein System ausreichende Signalqualität und geeignete Algorithmen aufweist, kann eine Portierung zu anderen erfolgen, wie Tablet oder Steuerungen von Industrieanlagen, erfolgen. In dieser Arbeit werden die Grundlagen geschaffen um das EOG Signal mit Hilfe des Dipol-Modelles auszuwerten. Die Arbeit befasst sich mit messen, übertagen und speichern des EOG -Signals. Sowie mit der Interpretation der gemessenen Daten mittels des Dipol-Modells. Der zu entwickelnde Algorithmus soll die grundlegenden Augenbewegungen und Augenpositionen erkennen.

1.2 Aufgabenstellung

Diese Arbeit beschäftigt sich einerseits mit der Weiterentwicklung eines Biosignalerfassers für Biosignale, der mehrere Kanäle anbietet, und andererseits mit der Verarbeitung des EOG Signals mit Hilfe des Dipol-Modelles. Das EOG Signal besteht aus mindestens einem horizontalen und einem oder zwei vertikalen Signalen. Je nach Anzahl von Elektroden können auch zwei horizontale sowie zwei vertikale Signale gemessen werden. Häufig wird auch das horizontale Signal über beide Augen gemessen, während das vertikale Signal bei jedem Auge separat gemessen wird. Daher sind mindestens zwei Kanäle am Biosignalerfasser erforderlich. Mit Hilfe des Dipol Modelles sollen pro Kanal drei Augenpositionen aus den erfassten Biosignalen erkannt werden. Sobald eine rasche Augenbewegung zu diesen Positionen erfolgt, soll ein Event generiert werden. Zum anderen soll die aktuelle Augenposition auf ein zweidimensionales kartesisches Koordinatensystem dargestellt werden. Damit sollen Grundlagen erarbeitet werden um Steuersignale für Computer zu ermöglichen. Die Auflösung und die verwendete Abtastrate muss ausreichend für EOG Signale gewählt werden. Die Bedienung des Gerätes soll möglichst einfach gehalten werden und intuitive erfolgen. Mit Hilfe einer Applikation auf einem mobilen Gerät, wie Smartphone oder Tablet, sollen die Parameter der Biosignalerfassung eingestellt werden. Da gängige Biosignalerfasser eine Vielzahl von Einstellungen bieten, sollen diese möglichst einfach auf dem verwendeten Tablet oder Smartphone einstellbar sein. Um die Einstellungen der Biosignalerfassung zu ändern, muss die Firmware auf dem Biosignalerfasser erweitert werden. So soll die bestehende unidirektionale Verbindung der seriellen Schnittstelle zu einer bidirektionalen Schnittstelle erweitert werden. Auf dieser Schnittstelle wir ein Protokoll modelliert und implementiert, damit die möglichen Parameter für den Biosignalerfasser einstellbar sind. Um nicht von der Applikation am Tablet oder Smartphone abhängig zu sein, soll das zu entwickelnde Protokoll ein Klartextprotokoll sein, damit auch eine einfache Steuerung über eine simple Terminalverbindung möglich ist. Damit ist es auch möglich von anderen Gräten, wie einem PC, die Einstellungen leicht zu ändern. Der gemessene Signalverlauf soll in Echtzeit dargestellt, sowie gespeichert werden können. Die Speicherung soll sowohl auf dem Messgerät, als auch auf dem Steuergerät erfolgen. Um verschiedene Datensätze unterscheiden zu können, soll mittels Zeitstempel eine automatische File-Benennung möglich sein. In

TU Bibliothek Your knowledge hub WIEN
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

der Smartphone- oder Tablet-Applikation soll eine grafische Anzeige implementiert werden. Es sollen zwei verschiedenes Graphen erstellt werden. Ein Graph soll ein Signal über die Zeit visualisieren, der zweite Graph die beiden EOG Signale orthogonal zueinander darstellen. Anhand des abgetasteten EOG Signals soll ein Algorithmus für das Dipol-Modell für Augenbewegungen entwickelt werden. Dabei sollen pro Signal drei unabhängige Augenpositionen erkannt werden. Bei einer raschen Blickänderung zu einer dieser Positionen, soll einem Event generiert werden. Weites soll noch die Blickrichtung in einem XY-Koordinatensystem angezeigt werden. Diese Steuersignale sollen in erster Linie auf einem Computer verarbeitet werden können. Bei der Programmierung ist dabei zu achten, dass es möglich sein soll, eine Dynamic Link Library (DLL) zu erstellen, die in einem weiteren Schritt in andere Programme eingebunden werden kann. Die DLL stellt die Augenpositionen als auch die Events für andere Programm zu Verfügung.

Die Arbeit wird im Rahmen von den Forschungsprojekten "KlaVig" und "LiveEOG" am Austian Institute of Technology (AIT) erstellt. Beide Projekte werden durch die Österreichische Forschungsfördergesellschaft (FFG) gefördert. Der Biosignalerfasser wurde im Rahmen des Projekt „KlaVig“ vom AIT entwickelt. Für diese Arbeit wird ein Gerät bereitgestellt, welches bereits die Hardwareanforderungen für diese Arbeit erfüllt. Das Gerät erfüllt alle regulatorischen Richtlinien um bei klinischen Prüfungen eingesetzt werden zu können und ist nach Klasse IIa eingestuft. Im Anhang befinden sich die Schaltpläne des übernommenen Gerätes. Alle Änderungen, die im Rahmen dieser Arbeit gemacht wurden sind ausgewiesen. Die Hauptaufgabe besteht in der Weiterentwicklung der Firmware, verbunden mit Modellierung und Implementierung eines geeigneten Übertragungsprotokolls für Daten und Steuerung des Gerätes. Die Funktion des Gerätes soll weitgehend nicht verändert werden. In Abbildung 1-1 ist eine schematische Darstellung des Biosignalerfassers dargestellt. Bei der Anwendung

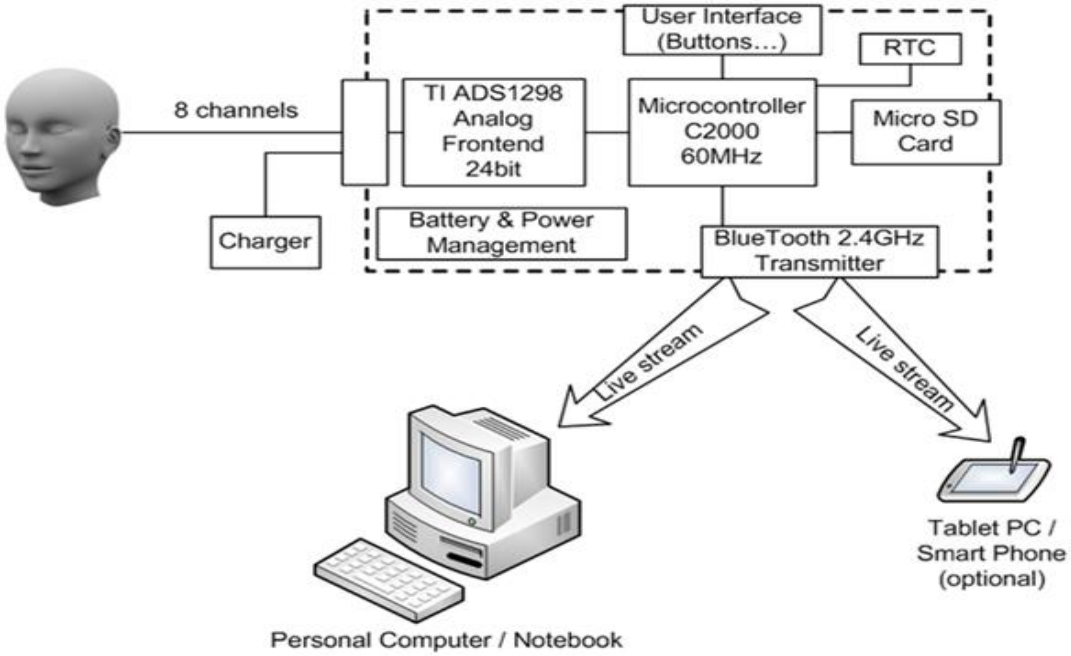


Abbildung 1-1 schematische Darstellung Biosignalerfassers (KlaVig Device, AIT)

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

besteht direkter galvanischer Patientenkontakt über das Anwendungsteil „Elektrodenkabel“ mit 5 Polen. Dabei ist der erste bipolare Kanal für die Messung des horizontalen Signals vorgesehen und der zweite Kanal für die Messung des vertikalen Signals. Der fünfte Pol wird als gemeinsam genutzte Referenzelektrode verwendet. Für diese Arbeit werden von den 8 Kanälen nur die ersten beiden Kanäle für das EOG-Signal verwendet. In dem Biosignalerfasser kommt das Analog Frontend ADS1298 von Texas Instruments (TI) zum Einsatz. Dieser IC bietet eine Vielzahl von Einsatzmöglichkeiten und Optionen an. Die Android®-Applikation soll, die für diese Arbeit benötigten Register des Chips, steuern können. Als Microcontroller steht ein C2000 zur Verfügung. Über eine serielle RS-232 Schnittstelle können Daten an eine Bluetooth Device übertragen, sowie empfangen werden. Die Eingebaute RTC soll über das zu entwickelnde Protokoll synchronisiert werden können. Die Micro SD Card dient zur lokalen Speicherung der gemessenen und abgespeicherten Rohdaten und soll in dieser Arbeit nicht verändert werden. Wie aus der Abbildung ersichtlich ist, ist keine bidirektionale Kommunikation vom Personal Computer, Tablet oder Smartphone zum Biosignalerfasser möglich, dies soll im Rahmen dieser Arbeit geändert werden. Sämtliche Änderungen der Firmeware werden in dieser Arbeit beschrieben. Als einzige HID am Biosignalerfasser fungieren ein Schalter und ein Taster. Mit dem Schalter kann das Gerät lediglich ein und ausgeschalten werden. Der Taster startet die lokale Speicherung, wenn sich eine SD Karte im Gerät befindet. Wenn keine Speicherkarte im Gerät ist, wird die Streaming Funktion gestartet. Mit Hilfe dieses Biosignalerfassers werden verschiedene EGO Signale erfasst, um daraus einen Algorithmus für das Dipol-Augen-Modell zu entwickeln. Die Entwicklung des Algorithmus erfolgt im Forschungsprojekt „LiveEOG“ und soll die Grundlage erarbeiten um eine neue Hardware speziell für die Messung von EOG-Signalen in dynamischen Umgebungen zu entwickeln. Danach wird das Gerät verwendet um in Echtzeitanwendungen das Signal an den Computer zu übertragen um den Algorithmus in Echtzeit validieren zu können. In der Software werden die Signale analysiert und das Ergebnis zur Weiterverarbeitung bereitgestellt.

Folgende Sicherheitshinweise müssen bei der Verwendung des Biosignalerfassers berücksichtigt werden:

- Schließen Sie an den Anschlussmöglichkeiten nicht eigenmächtig Kabel an, die vom Hersteller nicht speziell deklariert sind!
- Achten Sie bei der Verlegung des Patientenkabels darauf, dass sich das Kabel nicht um den Hals des Probanden wickeln kann. Befestigen Sie die Kabel ggf. mit Tapes sicher am Körper.
- Kontrollieren Sie vor jeder Inbetriebnahme das Gerät und das Zubehör auf offensichtliche Schäden.
- Die Anwendung des Gerätes, insbesondere der Anwendungsteile, darf nur in Kontakt mit unversehrter Haut erfolgen. Elektroden dürfen nicht an offenen, wunden Hautstellen verwendet werden, dies bringt die Gefahr einer Infektion mit sich.
- Um das Risiko eines elektrischen Schlages zu vermeiden, darf dieses Gerät während der Signalerfassung NICHT am Stromnetz angeschlossen werden.
- Leitfähige Teile der Elektroden und damit verbundene Steckvorrichtungen sollten andere leitfähige Teile einschließlich Erdverbindungen nicht berühren.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

1.3 Methodisches Vorgehen

Gerade in der Medizintechnik ist das methodische Entwickeln von Software notwendig, um die Sicherheit für den Anwender zu gewährleisten. Dafür gibt es viele Normen und Guidlines. Als Beispiel sei nur auf die ISO 13485 [1] verwiesen, die das methodische Vorgehen und deren Dokumentation für die Entwicklung von Medizinprodukten beschreibt. In Abbildung 1-2 ist das klassische V-Modell ab-

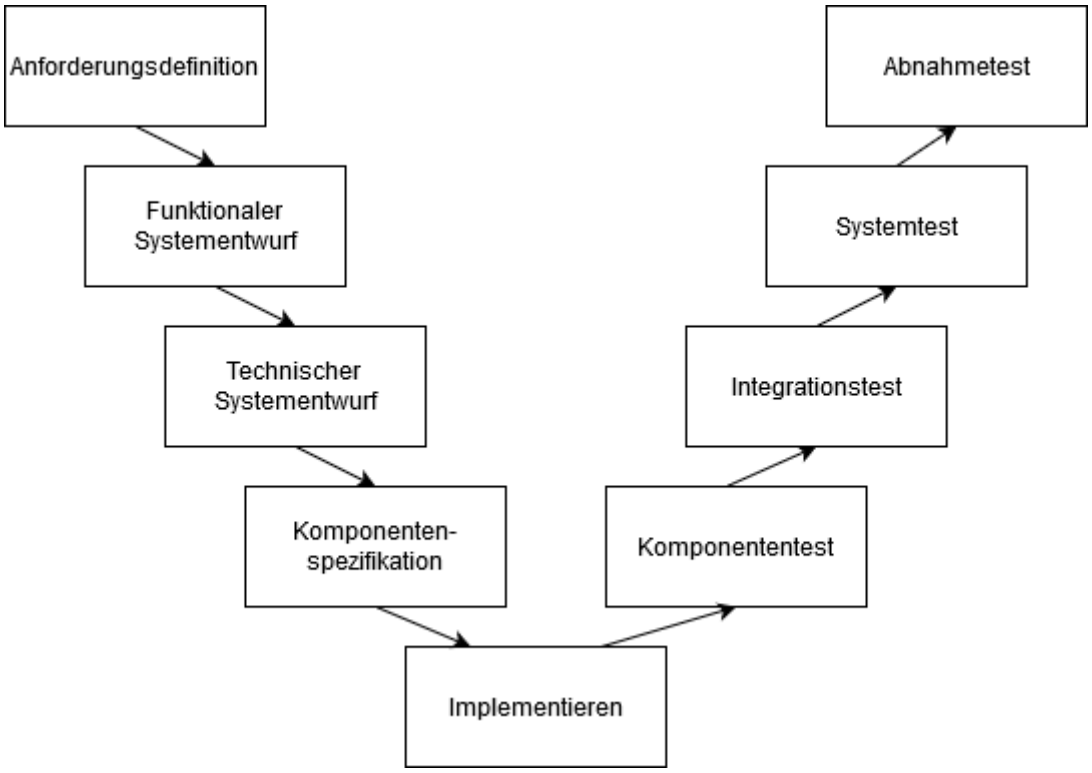


Abbildung 1-2: klassischen V-Modell

gebildet. Eine ausführliche Beschreibung kann z.B.: im Buch Das V-Modell 97 [2] gefunden werden. Das V-Modell beschreibt auf der linken Seite eine immer detailliertere technische Spezifikation und Implementierungsgrundlage. Nach der Implementierung, die Spitze des V-Modells, erfolgt auf der rechten Seite das Testen gegenüber der Spezifikation. Am Beginn des V-Modelles werden die Anforderungen an das zu entwickelnde System ermittelt. Eine gute Übersicht über das Ermitteln von Anforderungen kann in dem Buch von Klaus Pohl und Chris Rupp mit dem Titel: Basiswissen Requirements Engineering [3] gefunden werden. Diese Anforderungen werden nach und nach in technische Spezifikationen immer feiner umgesetzt damit sie dann implementiert werden können. Vor allem in Kapitel 6 werden die wichtigsten Komponenten der Software im Detail beschrieben. Im darauf folgenden Kapitel wird dann vor allem auf die Verifikation und Validierung der verschieden Softwaren eingegangen, die in dieser Arbeit entwickelt werden. Da in dieser Arbeit kein fertiges Medizinprodukt entwickelt wird, wird lediglich die Methodik übernommen und auf die wesentlichen Teile dieser Arbeit angewendet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

1.4 Struktur der Arbeit

Diese Arbeit gliedert sich grob in zwei Teile. Der erste Teil ist ein theoretischer Teil, in dem zuerst die Anatomie des Auges erläutert wird, danach wird auf die Entstehung des EOG-Signals eingegangen. Im Kapitel 3 wird eine kurze Übersicht, über wissenschaftliche Publikationen zur Auswertung von EOG-Signalen, gegeben. In diesem Kapitel werden auch kurz die Werkzeuge vorgestellt, die in dieser Arbeit zum Entwickeln der Software verwendet wurden. Auf die technischen Hintergründe der verwendeten Messkette, der verwendeten Übertragungstechnik sowie der benötigten Signalverarbeitung wird in Kapitel 4 eingegangen. Der zweite Teil der Arbeit beschäftigt sich mit der praktischen Umsetzung der Aufgaben. Es wird mit einer Beschreibung des zu entwickelten Algorithmus in Kapitel 5 begonnen, danach folgt die Implementierung in Kapitel 6 und in Kapitel 0 wird die Software getestet, verifiziert und validiert. Am Schluss der Arbeit werden die wichtigsten Ergebnisse zusammengefasst und ein Ausblick auf die nächsten nötigen Schritte gegeben.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

2. Anatomie des menschlichen Auges

Diese Arbeit beschäftigt sich mit der Messung, Verarbeitung und Auswertung des Elektrookulographie, daher wird ein elektrisches Signal gemessen dessen Ursprung das Auge bildet. In diesem Kapitel wird zuerst die Anatomie des Auges beschrieben, damit die Entstehung des Signals erklärt werden kann. Anschließend werden die Signaleigenschaften und auf Vor- bzw. Nachteile bei der Messung des EOG-Signals besprochen.

2.1 Aufbau des Auges

Das Auge ist eines der fünf Sinnesorgan und diehnt der Wahrnehmung optischer Informationen. Der anatomische Aufbau des menschlichen Auges ist nach [4], [5] und [6] verfasst worden.

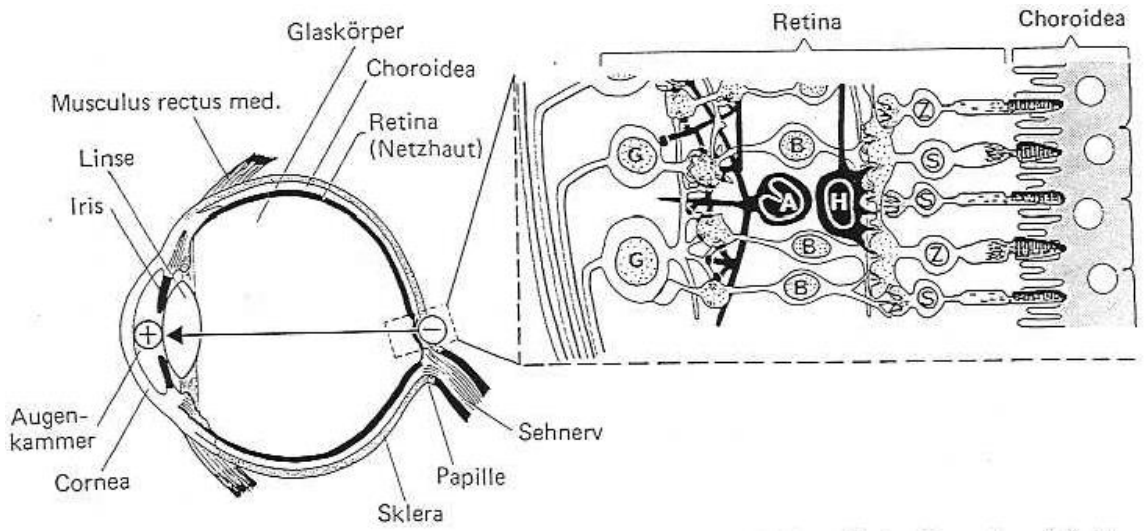


Abbildung 2-1: Anatomischer Aufbau des Auges. **[6]**

Eine schematische Abbildung des Augapfels wird in Abbildung 2-1 dargestellt. Der Augapfel ist kugelförmig, an der Vorderseite befindet sich die Hornhaut (Cornea). Die Regenbogenhaut (Iris) liegt zwischen der Cornea und der Linse. Diese drei Teile des Auges begrenzen die vordere Augenkammer. Hinter der Linse liegt die hintere Augenkammer, gefolgt vom Glaskörper der das Augeninneren bildet. In der vorderen und der hinteren Augenkammer befindet sich das lichtbrechende Linsensystem des Auges. An der Rückwand des Augapfels ist das lichtwahrnehmende System. Durch Verformung der Linse wird das scharf Sehen ermöglicht. Sie ist bikonvex und kann daher den Brechungsindex des Augapfels verändern. Gegenüber der Pupille befindet sich die Netzhaut (Retina). Sie beinhaltet die lichtempfindlichen Sinneszellen. Vor den Sinneszellen befindet sich noch das abdunkelnde retinale Pigmentepithel. Zwischen diesen beiden Schichten der Netzhaut bildet sich eine Potentialdifferenz

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

aus. Diese wird als Elekroretinographie (ERG) bezeichnet und bildet die Grundlage für das EOG Signal. Die Nerven der Netzhaut werden an einer zentralen Stelle der Papille zusammengefasst. Unterhalb davon befindet sich der Sehnerv, der die Information an das Kleinhirn weiterleitet. Die beiden Augenkammern werden vom sogenannten Kammerwasser ausgefüllt, welches eine wasserklare Flüssigkeit ist. Eine geleeartige Substanz, die größtenteils aus Wasser besteht, füllt den Glaskörper aus. Der Glaskörper ist unter Druck, damit der notwendige Kontakt zwischen Netzhaut und Aderhaut entsteht. Dieser Druck trägt zur Formgebung bei. Die Lederhaut (Sklera), die Aderhaut (Choroidea) und die Netzhaut - Retina genannt - bilden die Wand des Augapfels. Der Augapfel ist durch Verschiebungsplatten in einem Fettgewebskörper befestigt. Er ist nach allen Richtungen beweglich. Dies wird durch 6 Muskeln ermöglicht, wovon 4 gerade und 2 schräg ansetzen. Die geraden Muskeln bilden einen trichterförmigen Ring um den Augapfel und setzen an der Lederhaut vor dem Äquator an. Der rechte Muskel heißt „rectus medialis“, der untere Musekel ist der „rectus inferior“, der linke Muskel der „rectus lateralis“ sowie der obere Muskel „rectus superior“. Die schrägen Augenmuskeln setzen dagegen dahinter an. Diese beiden Muskeln heißen „obliquus superior“ und “obliquus inferior“. Die schrägen Muskeln sind für das Rotieren der Augen zuständig.

Die Augenbewegungen lassen sich auf Rotationen um drei Achsen in einem rechtwinkligen dreidimensionalen Koordinatensystem reduzieren. Dabei ist immer jeder Augenmuskel in einem Zustand der Entspannung oder der Kontraktion. Jeder Muskel ist bei jeder Bewegung dabei als Agonist oder als Synergist tätig. Bei einer Rotation des Auges um eine Achse erfährt der Agonist eine Kontraktion, während sein Gegenspieler der Antagonist entspannt wird. Diese wird im Gesetz der reziproken Innervation von Sherrington beschrieben [7]. Als Duktion bezeichnet man eine Drehbewegung des Auges. Als Vertikalduktion wir die Drehung um die horizontale x-Achse bezeichnet, bei einer Drehung um die y-Achse spricht man von einer Zykloduktion. Als Supraduktion wird die Bewegung nach oben bezeichnet, die Gegenbewegung wird Infraduktion genannt. Von einer Version spricht man, wenn die Augen konjugierte Bewegungen in die gleiche Richtung und um dieselbe Achse durchführen. Eine disjugierte Bewegung der Augen wird Vergenz genannt. Die Nullstellung beider Augen wird als Primärposition bezeichnet. Von dieser Position aus werden alle Augenbewegungen beschrieben. Sie darf jedoch nicht als Ruhepostion der Augen angesehen werden. Als Kardinalbewegungen werden Augenbewegungen aus der Primärposition bezeichnet, die nach oben, unten, links oder rechts gehen.
Die Steuerung der Augenmuskeln erfolgt durch das Hirn, sowie durch das Gleichgewichtsorgan im Innenohr. Damit wird das scharf Sehen genauso gesteuert, wie das Beobachten von Objekten selbst, wenn sich der Kopf bewegt und das Objekt nicht. Damit das Objekt beobachtet werden kann, muss das Objekt im Bereich des gelben Fleckes auf der Netzhaut abgebildet werden. Damit das der Fall ist, bewegen sich die Augen meistens synchron zueinander. Diese Bewegung wird durch das Gesetz von Hering beschrieben [8]. Das menschliche Sehen kann mit 5 Typen von Augenbewegungen beschrieben werden.

- Sakkaden
  Eine schnelle Augenbewegung wird als Sakkade bezeichnet. Sie kann Winkelgeschwindigkeiten von mehr als 600Grad pro Sekunde erreichen und dauert ca. 200ms. Sie dient dazu neue Objekte im Sichtfeld zu fixieren. Die Präzision einer Sakkade ist sehr hoch und erreicht eine Genauigkeit von ±1°. Sobald eine Sakkade ausgelöst worden ist, kann sie nicht mehr verändert oder unterbrochen werden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

- Folgebewegungen
  Mit Hilfe der Folgebewegung kann eine Fixierung eines bewegten Objektes erfolgen. Die Augenbewegung erfolgt mit ca. 1-30° pro Sekunde und ist sehr präzise. Sie ist eine Folge der Bewegung des Bildes im Bereich des Gelben Fleckes.
- Vergenzbewegungen
  Vergenzbewegungen dienen dazu die Augen so auszurichten, dass das fixierte Objekt auf der korrespondierenden Netzhaut beider Augen abgebildet wird. Die Geschwindigkeit beträgt ca. 6-15°/s. Sie treten z.B. auf, wenn sich die Entfernung zu dem Objekt ändert. Bei dieser Bewegung werden die Augen gegengleich bewegt.
- Vestibulärer Nystagmus
  Diese Augenbewegungen werden durch das Gleichgewichtsorgan im Ohr ausgelöst, die bei Kopfbewegungen entstehen, es handelt sich dabei um eine oszillierende Bewegung der Augen und dient zur Stabilisierung der Blickrichtung.
- Optokinetischer Nystagmus
  Hierbei handelt es sich um eine langsame Augenfolgebewegung, die durch eine relative Bewegung des Beobachters zu dem beobachteten Objekt ausgelöst wird.

Bei der Fixation eines Objektes treten Mikrobewegungen auf, die dazu führen, dass das Objekt nie lange auf derselben Stelle der Retina steht. Würde dies der Fall sein, wäre das Objekt nach kurzer Zeit aus der Wahrnehmung verschwunden.
Zu einer weiteren unkontrollierten Augenbewegung kommt es bei einem Lidschlag. Er verhindert das Austrocknen des Augapfels und dient nur zu Benetzung mit Tränenflüssigkeit. Das Blinzeln tritt alle 2 bis 10 Sekunden auf und kann stark streuen. Beim Blinzeln macht das Auge eine Supraduktion, die Stärke der Supraduktion ist dabei unterschiedlich.

2.2 EOG Signal

2.2.1 Signal Entstehung

Die Basis der Elektrookulographie bildet die Feldverschiebung des Elektrischen Dipols zwischen Cornea (positiv) und Retina (negativ) bei Augenbewegungen. Das Feld entsteht durch Ionentransport in den Pigmentschichten der gleichförmigen Ausrichtung der Zapfen und Stäbchen in der Retina. Zwischen dem negativen Pol an der Retina und dem positiven Pol der Kornea entsteht so eine Spannung von 0,4 bis 1 mV [9]. Dieses elektrische Feld lässt sich durch Elektroden, die in Augennähe angebracht sind abgreifen wie in Abbildung 2-2 a dargestellt wird. Die horizontalen Augenbewegungen werden binokulär, also als Summe der Potentiale an den äußeren Augenwinkeln, aufgezeichnet. Falls die horizontalen Augenbewegungen monokulär abgegriffen werden sollen, werden noch weitere Elektroden an den inneren Augenwinkeln angebracht. Mit Elektroden ober und unterhalb des Auges wird die vertikale Änderung gemessen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die Projektion des Cornea retinalen Dipols auf der Achse, die sich aus den jeweils gegenüberliegenden Elektroden ergibt, wird durch jede Augenbewegung verändert und somit die Potentialdifferenz in Abhängigkeit von der Augenposition. Dieses gemessene Potential entspricht dem Sinus des Winkels zwischen den elektrischen Achsen. Der Spannungsunterschied beträgt 15-20 µV pro Winkelgrad Augendrehung, bei einem linearen Messbereich von +/- 30°. In Abbildung 2-2b ist die Änderung des Potentials in Abhängigkeit der Blickrichtung dargestellt. Eine kompakte Analyse des EGO Signals kann auch in „EOG- gesteuerte Rechnersysteme mit Body-Area-Network-Anbindung“ [10] gefunden werden.

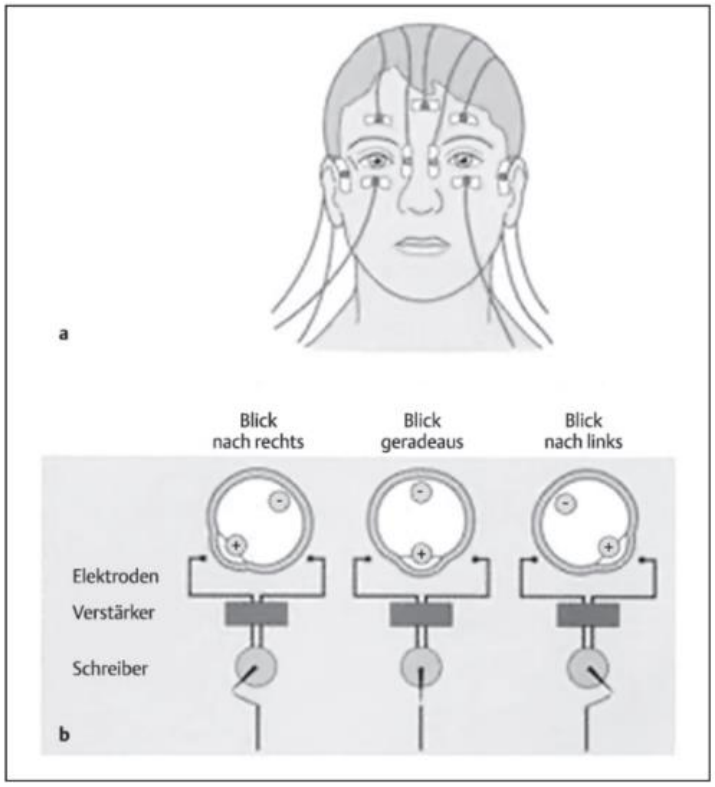


Abbildung 2-2: a) Position der Elektroden für EOG Messung**;** b) EOG Aufzeichnung von horizontalen Augenbewegungen. **[9]**

2.2.2 Signaleigenschaften

Die Potentialdifferenz beträgt zwischen 50 µV und 3500µV bei 15-20 µV pro Winkelgrad [11]. Das Signal deckt das gesamte Blickfeld von 180° in horizontaler Richtung, 60° nach oben und 70°, nach unten mit einer Genauigkeit von 1,5-2 Winkelgrad ab. Es ist sehr störanfällig von eingekoppelten Signalen, aber auch durch Artefakte von Muskelbewegungen oder Kopfbewegungen.

Um Augenbewegungen zu detektieren gibt es alternative Systeme wie das Kontaktlinsen / Search-Coil-Verfahren, die Photookulographie, die Videookulographie oder auch die Infrarotokulographie.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Eine kure Abhandlung der einzelnen Verfahren kann in der Arbeit von Sahra Brunner gefunden werden [12]. Im Gegensatz zu den genannten Methoden bietet das EOG einige Vorteile. Die Messung des EOG-Signals ist nicht invasiv und schränkt das Sichtfeld des Benutzers nicht ein. Brillen oder Kontaktlinsen beeinflussen die Messung nicht. Die Messung ist auch bei Kopfbewegungen oder geschlossenen Lidern möglich. Die Messung des EOG Signals hat auch Nachteile. Das Signal wird durch Artefakte gestört und hängt von der Umgebungshelligkeit ab. Die Artefakte können von Bewegungen, wie kauen oder Bewegungen im Gesicht, herrühren. Das EOG Signal kann auch von EEG bis hin zum EKG beeinflusst werden. Eine langsame Signaldrift entsteht durch die Veränderung der Leitfähigkeit der Haut unter den Elektroden, da man unter den Klebeelektroden zu schwitzen beginnt. Lidschläge führen zu Artefakten am vertikalen Signal.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

3. State of the Art

Dieses Kapitel widmet sich dem aktuellen Wissenstand auf dem Gebiet der EOG-Analyse, sowie etablierten Software-Werkzeugen zum Erstellen von Software und Algorithmen um die Signale aus der Messung auswerten zu können.

3.1 Auswertung von Elektrookulogramms

Das EOG Signal wird in der Wissenschaft für verschiedene Zwecke ausgewertet. So wird das EOG-Signal unter anderem verwendet für:

- Auswertung bezüglich Vigilanz oder Wachsamkeit gemeinsam mit EEG [13].
- Feststellung von Störungen des okulomotorischen oder vestibulären System [14].
- Aktivitätserkennung über Augenbewegungen wie lesen, fernsehen oder spazieren [15], [16]
- Erkennung einer Makulardegeneration [17]
- Generierung von Steuersignalen für computergestützte Systeme

Diese Arbeit beschäftigt sich ebenfalls mit der Generierung von Steuersignalen für computergestützte Systeme. Gemeinsam mit dem AIT hat Stan Hanke, in seiner Arbeit: EOG-gesteuerte Rechnersysteme mit Body-Area-Network Anbindung [10], sowie Sara Brunner in ihrer Arbeit: Entwicklung, Implementierung und Validierung der Signalverarbeitung für ein EOG-Kommunikations- und Steuerungssystem [12] mit dem Thema auseinandergesetzt. Mit Hilfe von Pattern Recognition wurde das EOG Signal auswertet. In [18] wird das EOG Signal ebenfalls mit Hilfe von Pattern Recognition ausgewertet. Es wird sowohl das vertikale, als auch das horizontale Signal verwendet. Durch das Erkennen der korrespondierenden Signalmuster für links, rechts bzw. oben und unten werden Steuersignale abgeleitet. In [19] wird nur das horizontale EOG Signal ausgewertet und mittels den drei Zuständen links, Mitte und rechts ein System beschrieben, dass es ermöglicht Wörter auf einen Bildschirm zu schreiben. Das System ist Kabelgebunden und für stationäre Benutzung konzipiert. In [20] wird das EOG Signal mit einer Abtastrate von 30 Hz digitalisiert. Dieses Signal wird kabellos auf den Auswertecomputer übertagen. Die Auswertung erfolgt über einen Zustandsautomaten mit fünf Zuständen. Mit Hilfe des entwickelten Systems kann ein Roboterarm vordefinierte Jobs ausführen. In [21] wird ein EOG System beschrieben, indem mit Hilfe von 7 Zuständen ein virtueller Parcours am Computer absolviert wird. Das System arbeitet mit einer Abtastrate von 256Hz. Der Auswerte-Algorithmus ist auf eine fest vorgegeben Testsetup zugeschnitten.

In dieser Arbeit sollen die Grundlagen erarbeitet werden um das EOG Signal unabhängig von der Umgebung oder Bewegung des Anwenders auswerten zu können. Die Erkenntnisse dieser Arbeit werden dann im FFG Projekt LiveEOG verarbeitet und wenn nötig Empfehlungen für die Verbesserung eines universell nutzbaren EOG-Devices umgesetzt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

3.2 Software-Werkzeuge

Integrierte Entwicklungsumgebungen (IDE) stellen eine Vielzahl von hilfreichen Werkzeugen zum Entwickeln von Software bereit. In den meisten IDEs sind Standartwerkzeuge wie Texteditor, Compiler, Linker oder ein Debugger integriert. In dieser Arbeit wird Code in drei verschiedenen Programmiersprachen entwickelt. Im Folgenden werden kurz die verwendeten Entwicklungsumgebungen beschrieben.

3.2.1 Android® Development Tool

Wesentlich zum Erfolg von Andoid[1] haben auch die mitgelieferten Entwicklungswerkzeuge beigetragen. Das Android Development Tool (ADT) bietet eine komplette Entwicklungsumgebung basierend auf einen Plug-in für Eclipse®. Es wird für Windows, Linux und Mac OS angeboten. Es können die verschiedene Version der SDK samt einen umfangreichen Emulator verwaltet werden. Durch die rasante Verbreitung der Android-Plattform seit ihrer Vorstellung 2007, ist es mittlerweile das am meist verwendete Betriebssystem für Mobile-Geräte [22]. Damit eignet sich Android ideal für diese Arbeit.

3.2.2 Code Composer Studio

Code Composer Studio ist eine Entwicklungsumgebung zur Entwicklung von Anwendungen für Mikrocontroller und eingebettete Prozessoren von Texas Instruments. Es enthält unteranderem einen optimierenden C /C++ -Compiler, eine Projekterstellungsumgebung, einen Debugger und einen Profiler. Das Tool wird in dieser Arbeit eingesetzt um die Firmware für den Mikrocontroller C2000 von TI zu erweitern. Die Verwendung des im Biosingalerfassers eingebauten Mikrocontroller macht den Einsatz von diesem Werkzeug notwendig.

3.2.3 MATLAB®

MATLAB2 ist eine kommerzielle Software und wird von MathWorks® hergestellt und vertrieben. Es wurde Ende 1970 an den Universitäten von New Mexico und Stanford entwickelt. Matlab ist die am weitesten verbreitete universell einsetzbare Mathematik-Programmiersprache, sowohl im akademischen als auch im kommerziellen Bereich.
Matlab ist spezialisiert auf einfache und schnelle nummerische Berechnungen, mithilfe von Matrizen und Vektoren. Matlab unterstützt die Verwendung von Skripten und benutzerdefinierten Funktionen, was beispielweise die Automatisierung der Verifizierung erlaubt.
Eine auf nummerische Matrizenoperationen spezialisierte Software ist ideal für die Verarbeitung von abgetasteten und quantifizierten Signalen geeignet. Durch diese Flexibilität und einfache Handhabung, sowie der Integrationsmöglichkeiten ist Matlab zu der Verbreitung gekommen die es heute in der digitalen Signalverarbeitung hat [23]. Vor allem die Möglichkeit eine DLL direkt aus Matlab zu erstellten ist dabei hervorzuheben. Matlab gehört sicher zur „State of the Arte“ Software in der digitalen Signalverarbeitung und wurde daher als Werkzeug für diese Arbeit ausgewählt.

[1] Android® ist eine eingetragene Marke der Firma Google LCC
[2] Matlab® ist eine eingetragene Marke der Firma MathWorks, Inc.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

4. Methoden

Diese Kapitel beschäftigt sich mit den Methoden die notwendig sind um bioelektrische Signale messen, übertragen und verarbeiten zu können.

4.1 Messung bioelektrischer Signale

Die aus der Messtechnik bekannte Struktur eines Messsystems gilt auch für die Messung bioelektrischer Signale und ist in Abbildung 4-1 dargestellt [24].

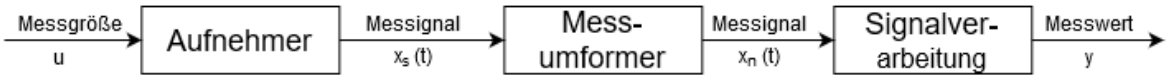


Abbildung 4-1 Messkette für bioelektrische Signale

4.1.1 Aufnehmer

Als Aufnehmer fungieren Ableitelektroden unterstützt von Elektrodenpasten, an dessen Eingang die zu messende Größe u anliegt. Das Messsignal kann in zwei qualitativ unterschiedliche Anteile zerlegt werden, dem Nutzsignal, welches der Potenzialdifferenz entspricht und dem Störsignal. Das Nutzsignal von Biopotenzialmessungen besitzt an der Körperoberfläche eine Amplitude von einigen Mikrovolt. Der menschliche Körper hat in der Regel keine Verbindung zu einem Bezugspotential, daher können Störungen über kapazitive Streukopplungen auftreten. Als Beispiel kann hier die Netzspannung angeführt werden. Das Störsignal kann bis zu 5 Größenordnungen größer sein als das Nutzsignal. Um bioelektrische Signale zu messen, die sich in Potentialdifferenzen wiederspiegeln, muss eine Verbindung zwischen den Menschen und dem Messgerät hergestellt werden. Dies erfolgt mit Hilfe von Elektroden. Die zumessende Potentialdifferenz entsteht durch eine Ionenleitung zwischen Retina und Korona. In Metallen beruhen die elektrischen Vorgänge jedoch auf Elektronenleitung. Elektroden bilden die Schnittstelle dazwischen. Es werden immer Potentialdifferenzen zwischen zwei Punkten gemessen. Eine detaillierte Beschreibung findet sich in [25].

Ableitelektroden

Die physikalische Struktur der Grenzschicht, zwischen dem Festkörper-Elektrolyt-System und der daraus resultierende Potentialverlauf, sind entscheidend für die verschiedenen Stromtransportmechanismen. An dieser Grenzfläche treffen zwei Systeme mit verschiedenen chemischen Potentialen aufeinander. Die Spannung zwischen den beiden Potentialen wird Galvani-Spannung [26] genannt die jedoch nicht direkt messbar ist. Der Ladungsaustausch zwischen festem Leiter und flüssigen Elektrolyt entsteht durch unterschiedlichen Austrittsarbeiten von Elektronen und Ionen aus beiden Medien. Diese Austauschreaktion bewirkt ein thermodynamisches Gleichgewicht, in dem das chemische Potential in jeder beteiligten Phase gleich groß ist. Es kommt zur Aufladung der Elektrode durch den Ausgleich

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

der chemischen Potentiale und damit zu elektrostatischen Wechselwirkungen zwischen den stark polaren Wassermolekülen und der geladenen Festkörperoberfläche. Diese Grenzschicht wird Helmholz-Doppelschicht genannt. Eine genaue Beschreibung der Grenzfläche kann in [25] gefunden werden. Jede Elektrode weist eine Strom-Spannungs-Kennlinie auf, die eine Beurteilung der chemischen Elektrodeneigenschaften, sowie des elektrischen Verhaltens wiedergibt. Im Bild a der Abbildung 4-2 ist ein schematischer Verlauf der Strom-Spannungs-Kennlinie bei Wechselspannungsmessung dargestellt wobei S die Stromdichte und ŋ die Überspannung ist. Als Überspannung ŋ wird die Differenz zwischen dem Elektrodenpotential und dem thermodynamischen Gleichgewicht bezeichnet. Zu jeder Elektrode gibt es auch ein Ersatzschaltbild welches in der Abbildung 4-2 b dargestellt ist.

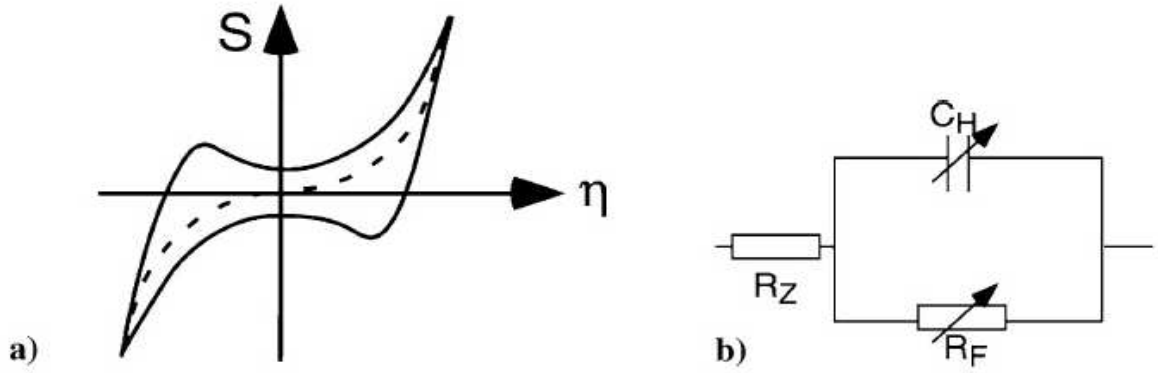


Abbildung 4-2: a) Strom-Spannungs-Kennlinie einer Elektrode b) Ersatzschaltbild einer Elektrode **[25]**

Es ist ersichtlich, dass jede Elektrode als Impedanz dargestellt werden kann, bei der die kapazitiven Effekte dominieren. R_Z gibt dabei den Zuleitungswiderstand an. Mit C_H wird die Übergangskapazität zur Haut beschrieben und R_F entspricht dem Übergangswiderstand zur Haut. Da die Galvani-Spannung nicht messbar ist, kann man nur Unterschiede messen. Daher muss eine zweite Elektrode, die je nach Messsystem platziert wird, verwendet werden. Als Messsystem können Monopolare oder bipolare Messungen der Signale verwendet werden. Bei monopolaren Messsystemen wird eine Elektrode an einem elektrisch inaktiven Punkt angebracht wie z.B.: Ohrläppchen. Bei bipolaren Messystemen werden beide Elektroden an elektrisch aktiven Punkten angebracht. Bei dieser Messung werden gleichphasige Signalanteile ausgelöscht und gegenphasige verstärkt. Hingegen wird bei monopolaren

Referenzelektrode	System	Potenzial gegen NHE
Normalwasserstoff-Elektrode (NHE)	Wasserstoff umspült Platinelektrode unter Standardbedingungen	0,000V
Kalomel-Elektrode	Hg/Hg_2Cl_2-Elektrode in gesättigter KCl-Lösung	0,241V
Siber-Silberclorid-Elektrode	Ag/AgCl-Elektrode in gesättigter KCl-Lösung	0,197V

Tabelle 4-1: Standardpotenzial wichtiger Referenzelektroden **[25]**

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Messystemen die Aktivität gegenüber dem Referenzpotential gemessen. Es gibt verschiede Elektroden aus verschieben Metallen. Tabelle 4-1 gibt eine Übersicht der häufigsten Elektroden Typen. Die Silber/Silberchlorid-Elektrode ist eine weit verbreitete Elektrode. In die Klasse der „vollkommen nicht polarisierbaren Elektroden“ wird diese Elektrode zugeordnet [27]. Sie besteht aus einem metallischen Silberkern. Dieser Kern ist mit Silberchlorid überzogen und befindet sich in einem Chlorionen gesättigten Elekrolyt. Sobald eine kathodische Belastung vorliegt, wandern Elektronen aus dem Metallkern in die Silberchloridphase und führen dort zum Zerfall von Silberchlorid und schließlich zur Desorption eines Chlorions. Im umgekehrten Fall, also bei einer anodischen Belastung der Elektrode, bestimmt die Anlagerung eines Chlorions und der damit verbundenen Freisetzung eines Elektrons, die Reaktion.

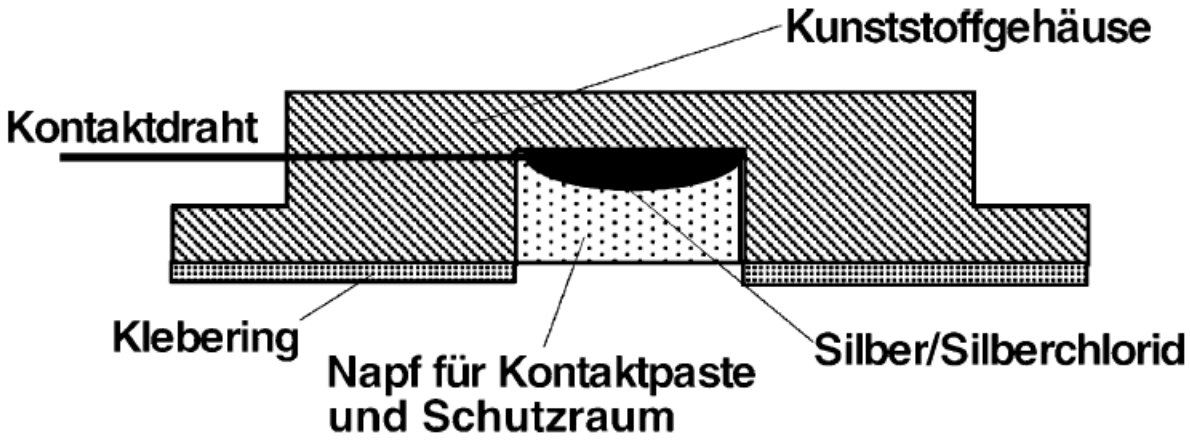


Abbildung 4-3: Schematischer Aufbau einer Silber/Silberchlorid-Elektrode. **[25]**

Abbildung 4-3 zeigt den Querschnitt einer Silber / Silberchlorid Elektrode. Um die Übergansimpedanz zwischen Elektrode und Haut zu verringern werden Elektrodenpasten eingesetzt. Damit wird die Übergangsimpedanz klein gegenüber der Eingangsimpedanz des Verstärkers. Bei Silber/Silberchlorid-Elektroden wird damit auch eine ausreichende Chlorionenkonzentration sichergestellt.

4.1.2 Elektrische Störquellen

Bei jeder Messung von Biosignalen gibt es verschiede Störquellen. Die Störquellen können sehr vielfältig sein. Einfluss auf die Messung haben sowohl mechanische Störquellen als auch elektrische Störquellen. Elektrische Störquellen haben verschiedenste Verursacher wie Industrie, Energieversorgung oder Übertragungstechnologien wie Funksignale. Die elektrischen Störquellen können nach der Art der Einbringung unterteilt werden [28]. So können Störungen galvanisch, kapazitiv, induktiv oder elektromagnetisch eingekoppelt werden.

4.1.3 Messumformer

Die Aufgabe in der medizinischen Messtechnik besteht darin Biosignale, die einige Mikrovolt bis Millivolt haben, zu erfassen und zu verstärken, [28]. Der Messumformer wandelt das gemessene Signal in ein geeignetes elektrisches Ausgangssignal um. So kann ein Messumformer Filter, Digitalisierung oder Übertragung beinhalten. Eine wesentliche Aufgabe des Messumformers ist die Störungen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

im Signal möglichst zu minimieren. Einen Teil des Störsignals bilden eingekoppelte Störspannungen. Diese liegt normalerweise an allen Punkten in gleicher Amplitude und Phase an. Daher kann das Differenzsignal gebildet werden um das gewünschte Nutzsignal zu bekommen. Als Differenz kann ein Bezugspunkt verwendet werden. Diese Gewinnung des Messsignals findet im Messsystem im Messumformerblock statt. Auch die Digitalisierung des zu messenden Signals wird in dem Block behandelt und wird durch den Übergang von $x_s(t)$ in $x_n(t)$ gekennzeichnet. In diesem Block können auch analoge Filter sowie Verstärker zum Einsatz kommen.

Die zeitkontinuierlichen Signale des analogen Messwertaufnehmers werden in eine für den Computer lesbare Form gebracht. Hierfür wird das Signal zu diskreten Zeitpunkten abgetastet und anschließend einem quantisierten Wert zugeordnet. Beide Verfahren verfälschen zusätzlich das zu messende Signal. Damit kein signalrelevanter Informationsverlust bei der Abtastung entsteht, legt das Abtastheorem von Shannon [29] die Abtastrate fest.

4.1.3.1 **Abtastung**

Bei einem kontinuierlichen Signal kann zu jedem beliebigen Zeitpunkt die Signalamplitude angegeben werden. Wenn zu diskreten Zeitpunkten die Signalwerte erfasst werden spricht man von Abtastung. Das Abtasttheorem (4.1) gibt an wie häufig dies geschehen muss um das Signal ohne Informationsverlust wieder herstellen zu können. Wobei τ die Dauer der Zeitspanne zwischen den Abtastzeitpunkten an-

$$\tau = \frac{1}{2 * f_{max}} \quad f_{abtast} > 2 * f_{max} \quad (4.1)$$

gibt. Mit f_{max} wird die maximal vorkommende Frequenz im Nutzsignal bezeichnet. Sie wird auch als Nyquis Frequenz bezeichnet [30].

4.1.3.2 **Quantisierung**

Wenn einem stetigen Eingangssignal x(t) diskrete Werte $x_n(t)$ zugeordnet werden, spricht man von Quantisierung. Dabei kommt es zum Quantisierungsfehler. In Abbildung 4-4 ist die Kennlinie eines Quantisieres dargestellt. Dieser kann kleiner oder gleich der halben Quantisierungsstufe sein. Der Fehler kann mit der Formel 4.2 berechnet werden.

$$|e_n(t)| = |x_n(t) - x(t)| \leq \frac{q}{2} \quad (4.2)$$

Die Abtastung und die Quantisierung wird im Analog-Digital Converter realisiert. Es gib verschiede Arbeitsweisen von ADC. Häufige sind z.B.: A/D-Nachlaufumsetzer, ADC mit sukzessiver Approximation oder auch der Delta-Sigma-Converter (ΔΣ ADC).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

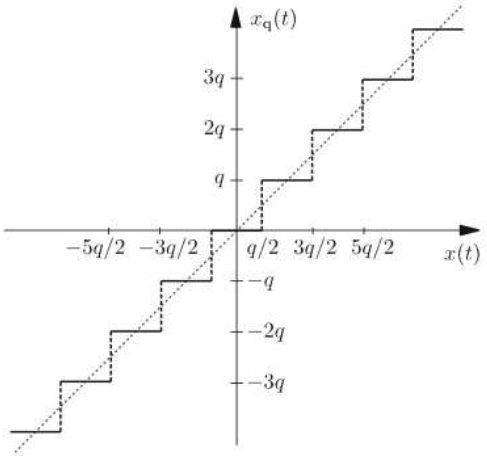


Abbildung 4-4: Kennlinie eines Quantisierers **[30]**

4.1.3.3 Delta-Sigma-Converter

Der Delta-Sigma-Converter zeichnet sich durch eine hohe Auflösung bei gleichzeitiger schneller Wandlung aus, das bei dynamischen Signalen Vorteile bringt. Typischerweise wird eine niedrige Auflösung von 1 Bit verwendet. Um die effektive Auflösung zu erhöhen, wird eine Überabtastung durchgeführt und damit kann auch das Quantisierungsrauschen reduziert werden. Er wird unter anderem in der Audiotechnik, der Konsumelektronik und der Kommunikationstechnik eingesetzt. Ein systematisches Blockschaltbild ist in Abbildung 4-5 dargestellt.

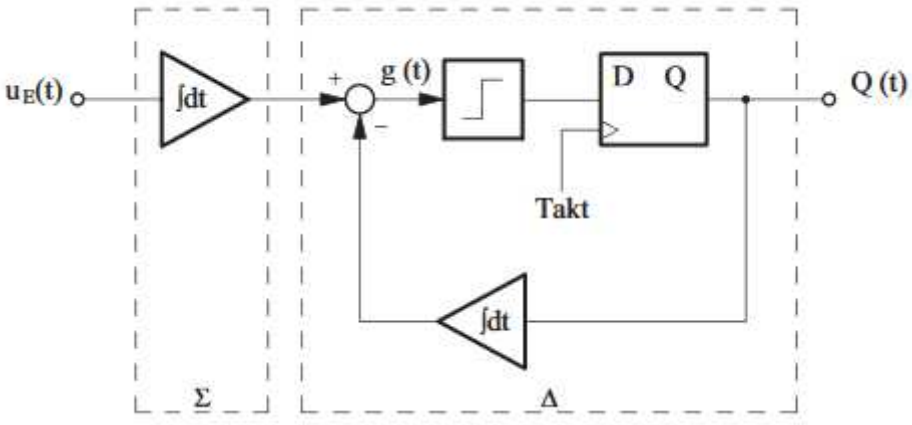


Abbildung 4-5: Blockschaltbild eines Delta-Sigma- Modulators **[29]**

Der Delta Modulator aus Abbildung 4-5 besteht aus einem Integrator, der Differenz zwischen dem Eingangssignal und dem zurückgeführten Ausgangssignal integriert. Das Eingangssignal wird zusammen mit dem rückgeführten Signal, einem Komparator mit nachgeschaltetem Schmitt-Trigger zugeführt. Der Schmitt-Trigger steuert ein Flip-Flop, welches eines der beiden digitalen Signale 0 oder 1 am Ausgang liefert. Wenn das Signal kleiner als die Ausgangspannung ist wird eine 0 ausgegeben, wenn das Signal größer ist, wird eine 1 ausgeben. Das zurückgeführte Signal wird durch zeitliche

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Integration des digitalen Ausgangssignals ermittelt. Damit auf das abgetastete Eingangssignal geschlossen werden kann, muss jedoch sehr hoch abgetastet werden, da die Differenz zum vorherigen Wert mit nur einem Bit quantisiert wird. Diese hohe Abtastung wir auch „oversampling“ genannt [30]. Der Delta-Sigma Modulator hat noch einen Integrierer der Delat Modulation vorgeschalten. Das setzt jedoch voraus, dass das Eingangssignal keine Gleichanteile enthält. Damit kann aus dem seriellen Bitstroms des Ausganges, durch Mittelwertbildung, direkt auf das analoge Eingangssignal geschlossen werden. Tiefergehende Beschreibungen und detaillierter Aufbau eines Delta-Sigma Converters kann in [30] oder [29] nachgeschlagen werden.

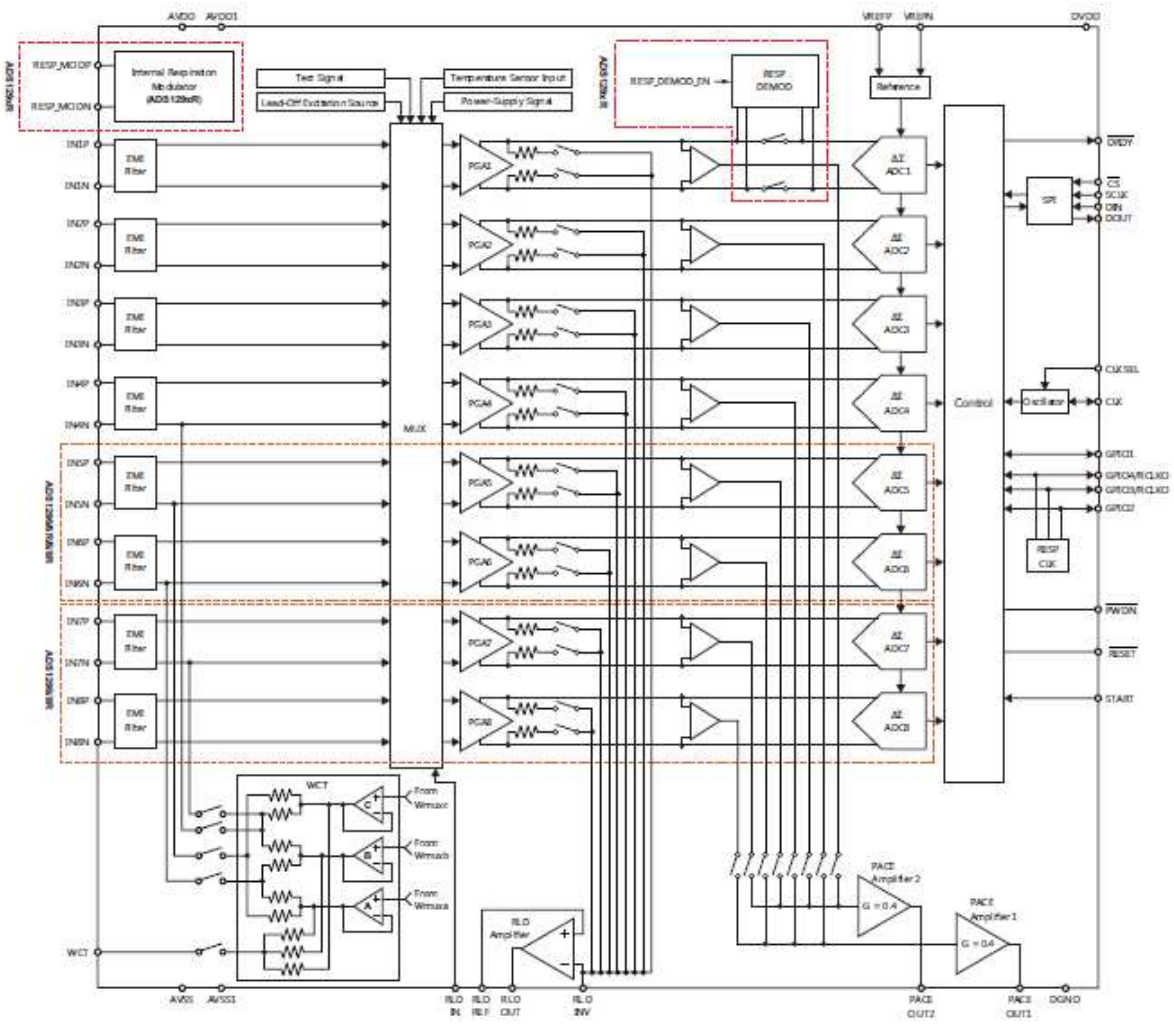


Abbildung 4-6 Funktionsblock Diagramm des ADS1298 von TI.

4.1.3.4 Analog Front-End für Biopotential Messungen

Bei der in diesem Projekt eingesetzter Hardware wird ein 24-Bit Analog-Front-End für Biopotential Messungen eingesetzt. Es handelt sich um den Chip ADS1298 von Texas Instruments. In Abbildung

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

4-6 ist das Funktionsblock Diagramm dargestellt. Zu erkennen sind die Eingangsfilter sowie die Sigma-Delta Wandler und die Verstärker, für jeden der 8 Kanäle. Dabei kann die Verstärkung für jeden einzelnen Kanal mittels Register gesetzt werden.

4.2 Datenübertragung

Wenn Daten auf einem Gerät vorliegen, diese aber bei einem anderen Gerät gebraucht werden, muss ein Informationsaustausch zwischen den Geräten stattfinden. Dieser Informationsaustausch wird Datenübertragung genannt. Ein Datenaustausch findet immer zwischen einem Sender und einem Empfänger statt. Damit beide Seiten sich verstehen, wird der Datenaustausch mit einem Protokoll geregelt.

4.2.1 Netzwerkprotokolle

[31] definiert ein Netzwerkprotokoll folgendermaßen: Ein Protokoll beschreibt einen genau definierten Kommuikationsprozess mithilfe von Regeln und Formaten, abhängig von der gestellten Aufgabe. Ein Protokoll definiert das Format und die Reihenfolge des Nachrichtenaustausches zwischen zwei oder mehr kommunizierenden Entitäten sowie die Handlung, die bei Übertragung und/oder Empfang einer Nachricht oder eines anderen Ereignisses ausgeführt werden.

Die Regeln hängen von verschieden Faktoren wie Übertragungsmedium oder Netzwerkaufbau ab. Damit nicht für jede Aufgabe ein neues Protokoll geschrieben werden muss, werden die unterschiedlichen Aufgaben einer Datenübertragung in sogenannten Protokollschichten unterteilt. Als Referenzmodell für eine Schichtarchitektur von Netzwerkprotokollen wird das Open System Interconnection Model (OSI) verwendet.

4.2.2 OSI Modell

Das OSI Modell beschreibt die Kommunikation zwischen zwei Systeme unabhängig von den Technischen Möglichkeiten. Es soll helfen die Komplexität von Telekommunikationsnetzen zu reduzieren, indem die einzelnen Funktionalitäten in verschiede Schichten aufgeteilt wird. Hierzu wurden sieben aufeinander folgende Schichten definiert. Jede Schicht beschreibt eine begrenzte Aufgabe sowie die Schnittstelle zu der nächsten Schicht. Die Schichten heißen: Physical Layer; Data Link Layer, Network Layer, Transort Session Layer, Presentation Layer und Application Layer. Jede Schicht nutz die darunter liegende Schicht und stellt der übergeordneten Schicht Ihre Funktionen unabhängig von den zu übertragenden Daten zur Verfügung. In Abbildung 4-7 sind die 7 Schichten des ISO-OSI Referenzmodelles mit den englischen und deutschen Namen dargestellt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

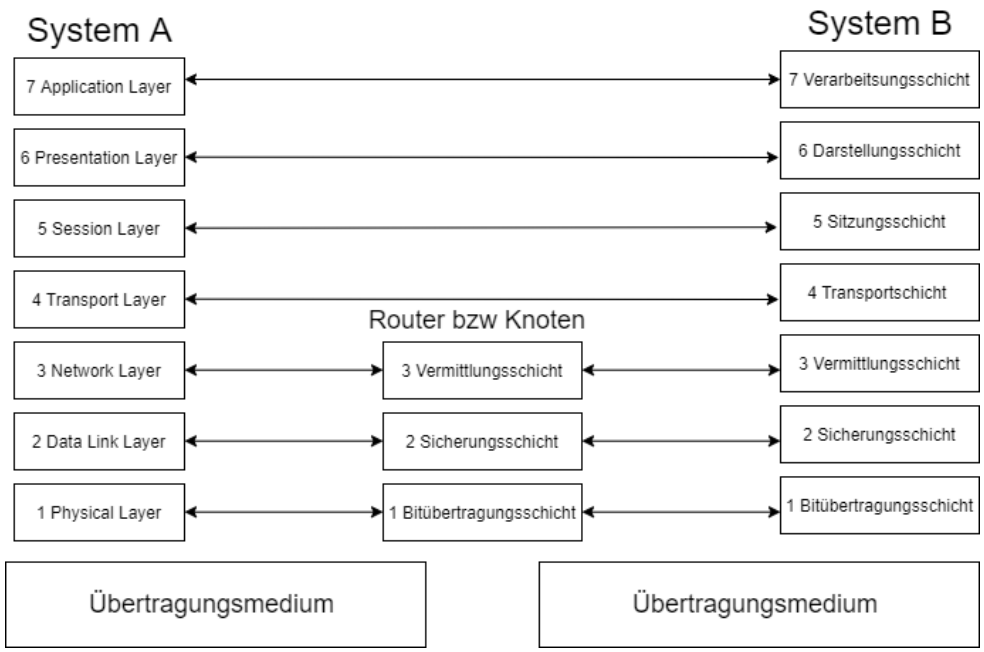


Abbildung 4-7: ISO-OSI Referenzmodell mit englischer und deutschen Bezeichnungen

Eine Übertragung zwischen zwei Knoten findet immer über die unterste Schicht statt. Diese wird auch Bitübertragungsschicht genannt. Die einzelnen Schichten tauschen Informationen immer nur mit ihrer Partner-Instanz aus, die auf der selben Schicht des Nachbarknoten befindet. Es besteht daher immer eine logische Verbindung zwischen den einzelnen Schichten. Der Informationsfluss geht jeweils durch alle darunter liegenden Schichten. Jede Schicht fügt ihren Header der Daten der darüberliegenden Schicht hinzu wie in Abbildung 4-8 zu erkennen ist. Alle Funktionen die für die Übermittlung einer Nachricht befinden sich in Schicht 1-3, diese Schichten sind auch für die Netzwerkbetreiber relevant. Die erste Schicht wird immer in Hardware implementiert. Die zweite Schicht wird meistens in Hardware implementiert. Schicht drei wir in Hochgeschwindigkeitsnetzen in Hardware und in allen anderen Netzen in Software implementiert. Die ersten drei Schichten werden auch durchlaufen, wenn das Übertragungsmedium geändert wird. Die Schichte 4-7 hingegen werden immer in Software implementiert und sind nur in den Endsystemen zu finden. Eine detaillierte Beschreibung des ISO-OSI Referenzmodell kann in [32] und eine gute Übersicht in [33] gefunden werden.

- **Schicht 1 - Physical Layer oder Bitübertragungsschicht**
  In dieser Schicht werden die elektrischen, optischen, elektromagnetischen Wellen oder mechanischen Eigenschaften beschrieben welche für die physikalische Verbindung zwischen den Systemen oder auch Entitäten nötig sind.
- **Schicht 2 - Data Link Layer oder Sicherungsschicht**
  In dieser Schicht werden Verfahren beschrieben um die fehlerfreie Übertragung der Nachricht zu gewährleisten. Dazu gehört die Zugriffsregelung auf das Übertragungsmedium oder die

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Fehlererkennung. Die hier aufgebaute Verbindung ist eine Punkt zu Punkt Verbindung, es werden immer Daten von einem Verbindungspunkt zum nächsten übertragen.

- **Schicht 3 – Network Layer oder Vermittlungsschicht**
  Die Vermittlungsschicht ist für das Routing (Wegsuche) verantwortlich. In dieser Schicht werden eindeutige Adressen in einem Netzwerk vergeben und die Datenpakete verteilt. Kommunikationsnetzte können aus unterschiedlichen Übertragungsmedien und den dazugehörigen Protokollen bestehen. In dieser Schicht werden die Übersetzungsfunktionen bereitgestellt. Ab dieser Schicht spricht man von einer End zu End Verbindung, da der Weg den die Daten gehen unbekannt sein kann.
- **Schicht 4- Transport Layer oder Transportschicht**
  Um die Segmentierung der zu übertragenden Daten kümmert sich die Transportschicht, damit es zu keinem Datenstau auf einzelnen Verbindungen kommt.
- **Schicht 5 – Session Layer oder Sitzungsschicht**
  Für den Auf und Abbau von Verbindungen zwischen zwei Endgeräten ist die Sitzungsschicht zuständig. Es werden sogenannte Fixpunkte (check points) definiert zu denen zurück gegangen werden kann wenn ein Verbindungsabbruch passiert.
- **Schicht 6 – Presentation Layer oder Darstellungsschicht**
  In dieser Schicht werden die Daten in eine nicht systemabhänige Form gebracht. Verschlüsselung und Datenkompression gehören auch zu den Aufgaben dieser Schicht.
- **Schicht 7 Application Layer (Darstellungsschicht)**
  Hier wird die Dateneingabe bzw. die Datenausgabe am jeweiligen System verwaltet. Der Auf- und Abbau aller unteren Schichten wird durch diese Schicht geregelt.

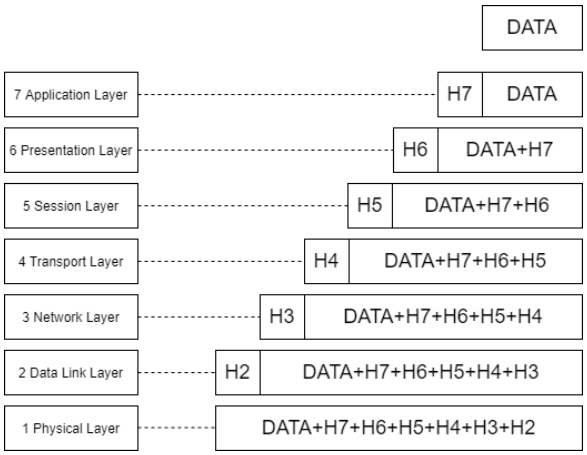


Abbildung 4-8: Den Nutzdaten wird bei jeder Schicht ein Header hinzugefügt

Obwohl das OSI Modell schon 1974 veröffentlich wurde- machen neuere offenere Netzwerktechnologien noch immer Gebrauch von der Idee und teilweise Gebrauch von Protokollen höherer Schichten.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

4.2.3 Bluetooth

Bluetooth ist eine Industriestandart für Datenübertragung per Funkt zwischen Gräten über kurze Distanzen. Die erste Version wurde im Juli 1999 veröffentlicht. Der aktuelle Standard Bluetooth 5 wurde 2016 freigegeben. Bluetooth wird von der Bluetooth Special Interest Group (SIG) weiterentwickelt. Das SIG „managment board“ ist in unterschiedliche Gruppen aufgeteilt, bei der jede Firma des Konsortiums mitarbeiten kann. Das Ziel des Konsortiums ist es eine offene Spezifikation für Sprach und Datenübertragung über eine kurze drahtlose Verbindung zu ermöglichen. Bluetooth sendet im lizenzfreien ISM-Band zwischen 2,402 GHz und 2,480 GHz.

4.2.3.1 Bluetooth Protokoll Stack

Ein wesentliches Merkmal der Bluetooth-Spezifikation ist, dass Geräte verschiedener Hersteller miteinander arbeiten können. Zu diesem Zweck definiert Bluetooth nicht nur ein Funksystem, sondern auch einen Softwarestack, mit dem Anwendungen andere Bluetooth-Geräte in der Umgebung finden und auch feststellen können welche Dienste diese anbieten. Der Bluetooth Stack ist in einer Reihe von Ebenen definiert. Es gibt einige Funktionen die mehre Ebenen umfassen. In Abbildung 4-9 ist der Bluetooth Stack im Vergleich mit dem OSI-Referenzmodell. In diesem Vergleich ist ersichtlich welche Ebene im Bluetooth Stack welche Verantwortung vom OSI-Referenzmodell übernimmt.

Der „Physical Layer“ ist für die elektrische Schnittstelle zu den Kommunikationsmedien verantwortlich, einschließlich Modulation und Kanalcodierung. Es deckt somit das Radio und einen Teil des Basisbands ab.

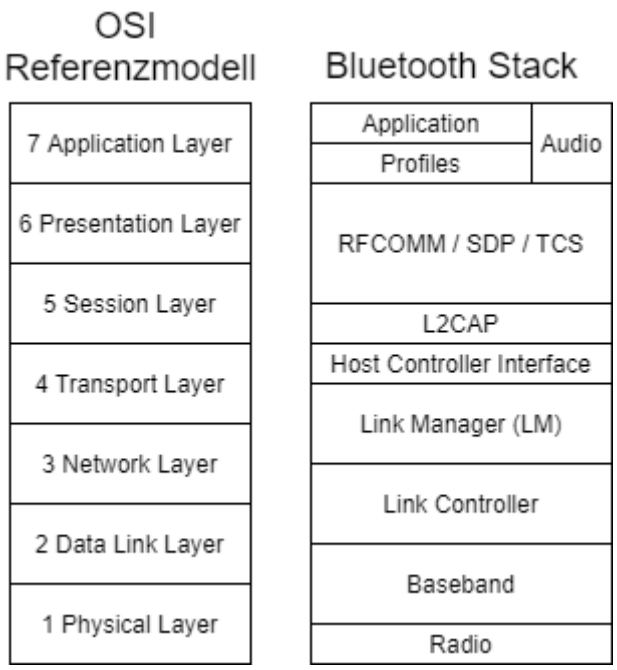


Abbildung 4-9: Gegenüberstellung OSI Referenz Modell zu Bluetooth Stack

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Der „Data Link Layer“ ist für die Übertragung, das Framing und die Fehlerkontrolle über eine bestimmte Verbindung verantwortlich und überlappt als solche die Verbindungssteuerungsaufgabe und das Steuerende des Basisbands, einschließlich Fehlerprüfung und -korrektur.

Der „Netzwerk Layer“ ist für die Datenübertragung über das Netzwerk verantwortlich, unabhängig von den Medien und der spezifischen Topologie des Netzwerks. Dies umfasst das obere Ende des Link-Controllers, das Einrichten und Verwalten mehrerer Links sowie den größten Teil der Link Manager-Aufgabe (LM). Der „Transport Layer“ ist für die Zuverlässigkeit und das Multiplexen der Datenübertragung über das Netzwerk auf die von der Anwendung bereitgestellte Ebene verantwortlich und überlappt sich somit am oberen Ende des LM und deckt die Host Controller Interface (HCI) ab, die den eigentlichen Datentransport bereitstellt Mechanismen.

Der „Session Layer“ bietet die Verwaltungs- und Datenflusssteuerungsdienste, die von L2CAP und den unteren Enden von RFCOMM / SDP abgedeckt werden. Der „Presentation Layer“ bietet eine allgemeine Darstellung für Daten der Anwendungsschicht, indem den Dateneinheiten eine Servicestruktur hinzugefügt wird. Dies ist die Hauptaufgabe von RFCOMM / SDP. Schließlich ist die Application Layer für die Verwaltung der Kommunikation zwischen Hostanwendungen verantwortlich. Hier biete Bluetooth verschiede Profile an. Für Audio Übertragungen biete Bluetooth einen eigenes Profil ab welches auch den Application Layer abdeckt.

4.2.3.2 **Bluetooth Kernprotokolle**

- Basisband

Die Basisbandschicht ermöglicht eine Funkverbindung zwischen Bluetooth-Geräten. Bluetooth das setzt das Frequenzsprungverfahren ein. Es wird daher zur Übertragung von Paketen ein definiertes Zeitfenster bei einer bestimmten Frequenz nutzt genutzt. Diese Schicht stellt die notwendigen Verfahren, mit denen die Übertragungsfrequenzen und der Takt der verschiedenen Geräte synchronisiert wird, zur Verfügung. Das Basisband stellt synchrone und asynchrone Verbindung zu Verfügung die im Multiplexverfahren dieselbe Funkverbindung nutzen können. Dabei wird die asynchrone Verbindung nur für Daten verwendet, bei der synchron werden auch Audio und Video Pakete versendet.

- Link Manager Protokoll

Für den Aufbau, Kontrolle und Steuerung der Verbindung sowie für das Aushandeln der Basisbandpaketgrößen ist das Link Manager Protocol (LMP) zuständig. Es beinhaltet außerdem die Funktionen für die Sicherheit der Verbindung wie Authentifizierung, Verschlüsselung oder Generierung, Austausch und Überprüfung der Verbindungs- und Verschlüsselungsschlüssel. Auch die diversen Energiemodi und Betriebszyklen der Bluetooth-Funkeinheit und der Verbindungsstatus der Einheit wird hier kontrolliert. Diese Schicht tauscht in eigenen LMP Paketen Informationen mit den anderen verbunden Bluetooth-Geräten aus. Diese Nachrichten haben eine höhere Priorität als Nutzerdaten.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

- Logical Link Control and Adaptation Protocol

Das Link Control and Adaption Protocol (L2CAP) unterstützt das Multiplexing der höheren Ebenen. Außerdem ist es für Paketsegmentierung und Paketneuszusammensetzung sowie Quality of Servive zuständig. Den höheren Schichten wird das Senden und Empfangen von Datenpaketen mit einer Größe von bis zu 64KB erlaubt. Das L2CAP ist definiert für asynchrone Verbindungen. Synchrone Verbindungen laufen rein über das Basisband.

- Service Discovery Protocol

Ein wichtiges Element innerhalb des Bluetooth System ist das Service Discovery Protocol (SDP). Es ist für die Diensterkennung zuständig und damit für sämtliche Einsatzmodelle. Diese Protokoll stellt, Geräteinformationen, Dienste und Leistungsmerkmale, aller vorhanden Dienste anderen Geräten zur Verfügung.

- Radio Frequency Communication Protokoll

Das Radio Frequency Communication Protokoll (RFCOMM Protokoll) emuliert eine serielle RS232 Verbindung über das L2CAP-Protokoll. Dabei werden Steuer – und Datensignale über das Basisband der Kabelgebunden Verbindung emuliert.

Daten zwischen Bluetooth Geräten werden mittels Profilen ausgetauscht. Eine Übersicht über gängige Profile kann in **Tabelle 4-2** gefunden werden. Die Liste ist bei weiten nicht vollständig zeigt aber den vielfältigen Anwendungsbereich von Bluetooth auf.

Abkürzung	**Bedeutung**	**Kurzbeschreibung**
A2DP	Advances Audio Distribution Profile	Übertragen von Audiodateien
BIP	Basic Imaging Profile	Übertragung von Bildern
BPP	Basic Print Profile	Drucken
DUN	Dial-up Networing Profile	Stellt eine Wählverbindung her z.B.: über ein Mobiltelefon
FAX	FAX Profile	Ermöglicht Faxen z.B.: über ein Mobiltelefon
HDP	Health Device Profile	Ermöglicht eine sicher drahtlose Verbindung zu medizinischen Geräten
HID	Human Interface Device Profile	Verdingung von HID Geräten wie Maus oder Tastatur
PBA	Phonebook Access Profile	Zugriff auf das Telefonbuch des anderen Gerätes
SPP	Serial Port Profile	Emuliert eine RS232 Verbindung über Bluetooth
VDP	Video Distribution Profile	Ermöglicht eine Übertragung von Videodatein

Tabelle 4-2: Beispiele für Bluetooth Profile

4.2.3.3 **Serial-Port-Profile**

Das Serial-Port-Profile (SPP) wird verwendet, wenn eine Kabelverbindung ersetzt werden soll. Es basiert auf dem allgemeinen Zugangsprofil (GAP), sodass sich über RFCOM serielle Kabelverbindungen nachbilden lassen. RFCOM wird für die Übertragung der Teilnehmerdaten, der Modem-Steuersignale und der Konfigurationsbefehle genutzt. Eine RFCOMM-Sitzung baut auf einen L2CAP-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Kanal auf. Abbildung 4-10 zeigt den verwendeten Protokoll Stack für eine SPP Verbindung. Bei einer Verbindung mittels SPP sind beide Geräte gleichberechtigt. Es gibt einen Initator von dem die Verbindung ausgeht und der Akzeptor nimmt die Verbindung an. In aktuellen Version Revison 12 der Spezifikation für das SPP Protokoll von Buetooth [34], werden Datenraten bis zu 250 kBit/s unterstützt. Es können mehre Verbindungen mittels SPP aufgebaut werden, auch zwischen gleichen Geräten. Bei mehreren Verbindungen können beide Geräte jede der beiden Rollen (Initator oder Akzeptor) übernehmen. Die Verwendung von Sicherheitsmerkmalen, wie Autorisierung, Authentifizierung und Verschlüsselung ist optional. Diese Option muss jedoch in dem Gerät implementiert werden damit die Funktion bei Bedarf genutzt werden. Verwendete Sicherheitsmerkmale werden beim Verbindungsaufbau einander zugeordnet und ändern sich nicht solange eine Verbindung über das SPP besteht. Die Bluetooth-SIG hat für die Emulgierung einer seriellen Kabelverbindung, zwischen zwei Geräten, drei Prozeduren spezifiziert.

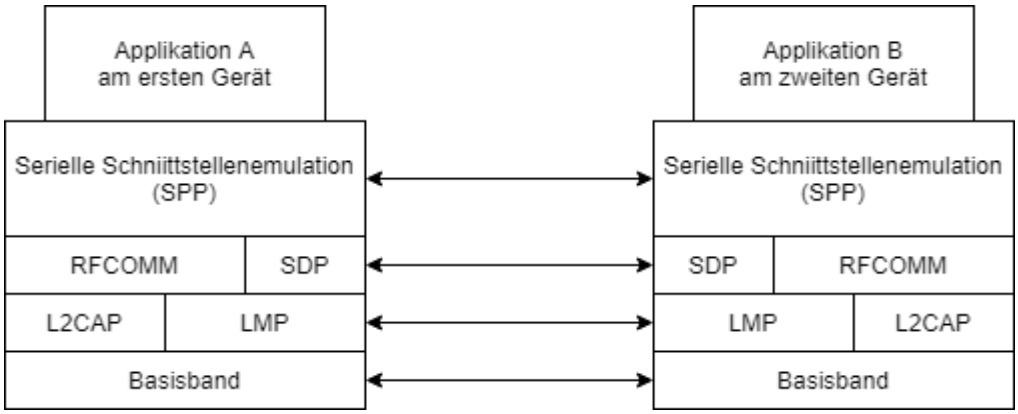


Abbildung 4-10: Protokollmodell für SPP

Bei der ersten Prozedur geht es um den Aufbau einer Verbindung. Sie besteht aus 4 Schritten und heißt: Establish Link/ Setup Virtual Serial Connection

1) Mit Hilfe des SDP wird eine Anfrage übermittelt. Die Anfrage wird mittels FRCOMM-Server-Kanalnummer bei der Anwendung des entfernten Gerätes durchgeführt. Der Teilnehmer kann eine der verfügbaren Schnittstellen (verfügbaren Dienst) beim anderem Gerät auswählen. Sobald der Teilnehmer mit einem bestimmen Dienst Verbindung aufnehmen will, muss er die entsprechenden Parameter mitschicken.
2) Optional muss sich das entfernte Gerät authentifizieren. Optional hat hier auch die Aktivierung der Verschlüsselung zu erfolgen.
3) Nun wird ein neuer L2CAP-Kanal vom entfernten RFCPOMM-Gerät angefordert.
4) Danach erfolgt die Initiierung einer RFCOMM-Sitzung auf dem L2CAP-Kanal.
5) Nun wird eine neue Datenverbindung über die RFCOMM-Sitzung über die Serverkanalnummer aus dem ersten Schritt erstellt.

Die zweite Prozedur wird auf dem anderem Gerät zeitgleich mit der ersten Prozedur ausgeführt. Damit hat sie ebenfalls 4 Schritte und heißt: Accept Link/Establish Virtual Serial Connection

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

1) Die allfällige angeforderte Authentifizierung leisten und die Verschlüsselung auf Aufforderung aktivieren.
2) Die Einrichtung eines neuen L2CAP-Kanals zulassen.
3) Die Einrichtung einer RFCOMM-Sitzung zulassen.
4) Die Datenverbindung über die RFCOMM-Sitzung zulassen.

Nach Abschluss der beiden beschreiben Prozeduren besteht eine virtuelle serielle Kabelverbindung zwischen den beiden Geräten.

Weiterführende Literatur zu Bluetooth kann in [35], [36] und [37] gefunden werden.

4.3 Signalverarbeitung

Die Signalverarbeitung hat zum Ziel, die Parameter aus dem Messsignal zu extrahieren (z.B.: Frequenz oder Amplitude.) Die Verarbeitung wird meist von Signalprozessoren übernommen oder einem darauf optimierten Halbleiterchips. Es gibt aber auch die Möglichkeit, dass die Signalverarbeitung vom Computern übernommen wird. Dazu muss das gemessene Signal hierfür auf das geeigneten Gerät übertragen werden. Signalverarbeitung beginnt schon bei der Vorbereitung zur Datenübertragung und endet erst dann, wenn die gewünschte Information in dem gewünschten Format bei dem verarbeitenden Gerät angekommen ist. Bei der Übertragung zu Computern wird die Vorbereitung und Codierung der Daten von einem Mikrocontroller übernommen. Diese gibt es bereits in fertigen Halbleiterchips, die unterschiedliche Funktionen anbieten. Die Endverarbeitung wird dann meist von einem Computer übernommen, der wesentlich mehr Rechenleistung bietet. Dieses Unterkapitel beschäftigt sich mit den in diese Arbeit verwendeten Signalverarbeiungsmethoden.

4.3.1 Medianfilter

Bei einem digitalen Filter liegt das Signal in der Form einer Liste mit numerischen Werten vor. Dieses Signal wir auch zeitdiskretes Signal und mit $x_{[n]}$ bezeichnet. Bei der Filterung werden die numerischen Werte in dieser Liste nach einer bestimmten Rechenvorschrift verändert.
Der Medianfilter gehört zu den nichtlinearen Filtern und wird in die Klasse der Rangordnungsfilter eingeordnet. Der Medianfilter entfernt „Außreißer“ in Messreihen [38]. Der Medianfilter ist definiert mit:

$$m(a_1, \dots \dots a_m) = die\ l - gr\ddot{o}\ss te\ Zahl\ der\ a_1, \dots \dots a_m \quad (4.3)$$

Dabei wird eine ungerade Messreihe umsortiert, nach steigender oder fallender Ordnung der Zahlen, der Wert der in der Mitte der umsortierten Reihe ist der Medianwert. Der größte Wert in der Liste wird mit dem Median ersetzt. Der Median Filter wird oft für die Entfernung von Störungen wie Rauschen eingesetzt. Bei der Anwendung des Medianfilter wird in der Signalverarbeitung ein Fenster definiert. Dieses Fenster gibt breite des Signals an auf das der Filter angewendet wird. Daher wird das Filter immer nur auf einen Teil des Signals. Das Fenster wird dann weitergeschoben und der Medianfilter wieder angewendet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

4.3.2 Zustandsautomaten

Als Automaten werden einfache Modelle für informationsverarbeitende Maschinen bezeichnete. Ein nicht deterministischer endlicher Automat lässt sich nach [39] folgendermaßen definieren:

Automat A =<I,O,X,R,f,g>

- I ist das Eingabealphabet
- O ist das Ausgabealphabet
- X ist eine endliche nicht leere Menge von Zuständen
- R ⊆ X ist die Menge der Anfangszustände und muss in X enthalten sein
- f ⊆ (X x l x X) heißt Übergangsrelation
- g ⊆ (X x I x O) heißt Ausgansrelation

Ein endlicher Automat heißt deterministisch wann es nur einen Anfangszustand gibt und wenn f und g Funktionen sind mit F : (X x I) →X und g : (X x I) → O.
Ein Automat wird mit Hilfe eines Zustandsdiagramms dargestellt. Dabei repräsentiert jeder Knoten einen Zustand und jede Kante einen Zustandsübergang. In Abbildung 4-11 ist ein Zustandsdiagramm

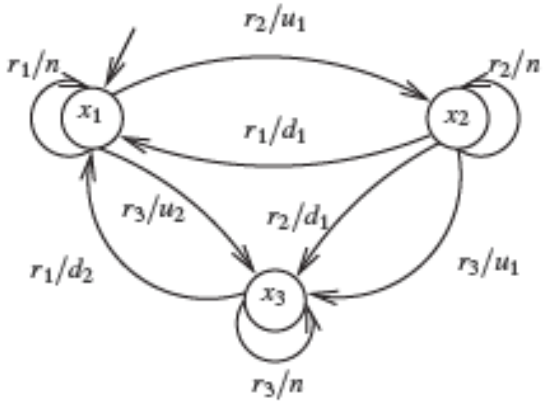


Abbildung 4-11: Zustandsdiagramm **[39]**

eines deterministischen endlichen Automaten dargestellt. Wenn das Eingabeereignis r_2 eintritt und der aktuelle Zustand X_1 ist, dann erfolgt ein Übergang zu X_2 und das Ausgabeereignis u_1 wird erzeugt. Bei dem dargestellten deterministischen endlichen Automaten sind { r_1 , r_2 , r_3 } das Eingabealphabet, n , u_1 , u_2 , d_1 , d_2 sind Elemente des Ausgabealphabet , die Menge der Zustände repräsentieren {x_1 , x_2 , x_3 }. Der Anfangszustand ist x_1 ist stabil.

Als ein Moore-Automat wird ein deterministischer, vollständig spezifiziert endlicher Automat bezeichnet, wenn alle Kanten r mit dem Endknoten x die gleiche Ausgabe o erzeugen.
Als Mealy-Automat wird ein deterministischer, vollständig spezifizierter endlicher Automat bezeichnet, wenn die Ausgabe o vom Zustand x und von der Kante r abhängen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

5. Modell für Augenbewegungen

In diesem Kapitel wird das verwendetet Modell für Augenbewegungen, mit Hilfe einer Analogie aus der Elektrotechnik beschrieben. Es soll dem Leser die Methodik erläutern wie mit der verwendeten Messmethode und Signalverarbeitung der implementierte Algorithmus funktioniert.

Das Auge fungiert als Dipol, wobei die Rückseite des Auges in Bezug auf die elektrisch positive Vorderseite des Auges elektrisch negativ ist. Dieses Phänomen wurde erstmals 1848 von Emil du Bois-Reymond beobachtet und ist ein Grundprinzip der Elektrookulographie. Ein Dipol Modell zu verwenden um Augenbewegungen zu beschreiben, ist daher eine übliche Methode, z.B.: in „Dipole models of eye movements and blinks“ [40] oder auch in „Elektrookulographie“ [14].

5.1 Elektrischer Dipol

Betrachtet man eine ideal leitende, homogen geladene Kugel mit dem Radus R, lasst sich die Ladung über das D-Feld berechnen $Q = \iint_O D.dA$. In Bezug auf den Raum außerhalb der Kugelekeltrode kann man die homogen geladene Kugel durch eine Punkquelle ersetzen. Mehrere Punktladungen können überlagert werden und ergeben ein Gesamtpotential in einem beliebigen Raumpunkt von

$$\varphi(r) = \frac{1}{4\pi\varepsilon}\sum_v \frac{Q_v}{\|r-r_v\|} \quad (5.1)$$

Dabei Entspricht Q_v die Ladung der Punktquellen und $\|r\text{-}r_v\|$ die Abstände der Punktquelle zum betrachteten Raumpunkt. Ein spezieller Fall ist der elektrische Dipol, er besteht aus zwei entgegengesetzten gleichgroßen Ladungen. Er wird charakterisiert durch den Abstand p und den Betrag der entgegengesetzten Ladungen Q. Die Abbildung 5-1 zeigt einen Elektrischen Dipol und seine elektrischen Feldlinien und die Äquipotentialflächen. Für einen elektrischen Dipol lässt sich das Potential auch in einem beliebigen Punkt folgendermaßen berechnen:

$$\varphi(\mathrm{P}) = \frac{1}{4\pi\varepsilon}\frac{p_{\mathrm{P}1}}{r_{\mathrm{P}1}^2} \quad (5.2)$$

Dabei ist zu beachten das die Entfernung des allgemeinen Punkts zum Mittelpunkt zwischen den Punkladungen groß gegenüber dem Abstand zwischen den Punktladungen ist. Mit p wird das elektrische Moment bezeichnet. Es ist eine Vektorgröße und fasst die beiden Punktladungen und ihre gegenseitige räumliche Lage zusammen [41]. Mit $r_{\mathrm{P}1}$ wird die Entfernung zwischen dem Ursprung des elektrischen Momentes und dem allgemeinen Punkt P bezeichnet. Eine wichtige Eigenschaft des elektrischen Momentes ist außerdem, dass eine Ladungsverteilung unabhängig vom Bezugsort ist, wenn die Gesamtladung gleich Null ist. Daraus folgt das der Ort der Bezugselektrode frei gewählt werden kann.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

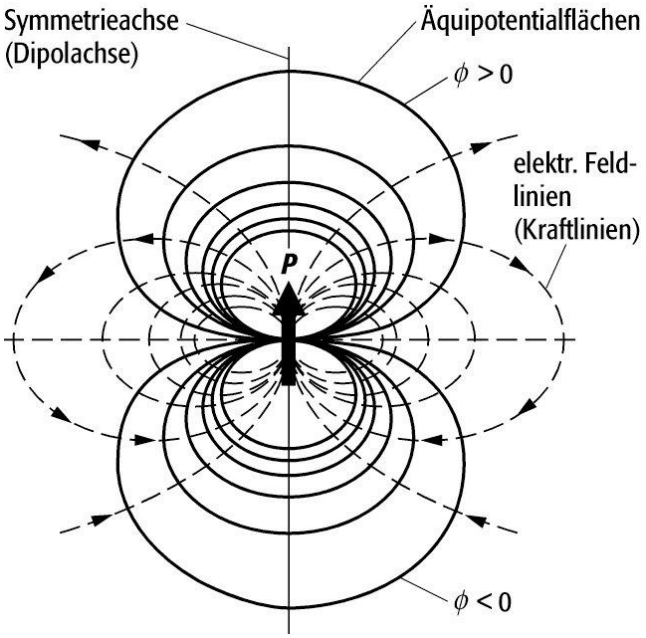


Abbildung 5-1 : Feldlinien eines elektrischen Dipols **[54]**

5.2 Dipol Modell für Augenbewegungen

Wie in Kapitel 2.2 ausgeführt entsteht das EOG Signal durch elektrische Dipole zwischen der Cornea (positiv geladen) und der Retina (negativ geladen). Die Summer der kleinen Dipole in den verschiedenen Abschnitten summieren sich zu einem Dipol in der Achse des Auges. Abbildung 5-2 zeigt im oberen Bild die Anordnung der Elektroden und im unteren Bild den dazugehörigen resultierenden elektrischen Dipol. Im Beispiel ganz links addieren sich die Potenziale und ergeben einen Dipol in der Mitte zwischen den Augen. Sollten die Elektroden so angebracht werden, dass sie nur den Dipol für ein Auge messen wie in dem mittleren Beispiel Abbildung 5-2 veranschaulicht ist, wird nur der Dipol eines Auges gemessen. Im rechten Beispiel, wird der resultierende vertikale Dipol dargestellt. Bei Messung von EOG Signalen wird von einer Grundlinie ausgegangen, dies sind die Auswirkungen des AC Verstärkers bzw. der Baseline Korrektur da der Gleichanteil entfernt wird. Bei Bewegung des Auges, ändert sich die Richtung des Dipols und damit das Elektrische Feld. Diese Änderung des Feldes, wird als EOG Signal aufgezeichnet. Daher wird nur das Ergebnis der Änderung des Dipols, wenn sich seine Position geändert hat aufgezeichnet. Die Dipole die wir beobachten und modellieren, sind Differenzdipole, die den Unterschied zwischen zwei Positionen des Augendipols darstellen. Daher kann jede Art von Augenbewegung von der Grundlinie weg unabhängig modelliert werden. Es kann erwartet werden, dass jeder Dipol eine andere Ausrichtung aufweist, die durch den Start- und den Endpunkt des Augendipols bestimmt wird. Aus dem Verhalten der Differenzdipole können wir Rück-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

schlüsse auf das Verhalten des Augendipols ziehen. Wenn das Auge gedreht wird, sollten die Differenzdipole nur ihre Ausrichtung ändern, nicht jedoch ihre Position. Wenn sich der Standort ändert, sollte dies auch der Unterschied der Dipole entsprechen.

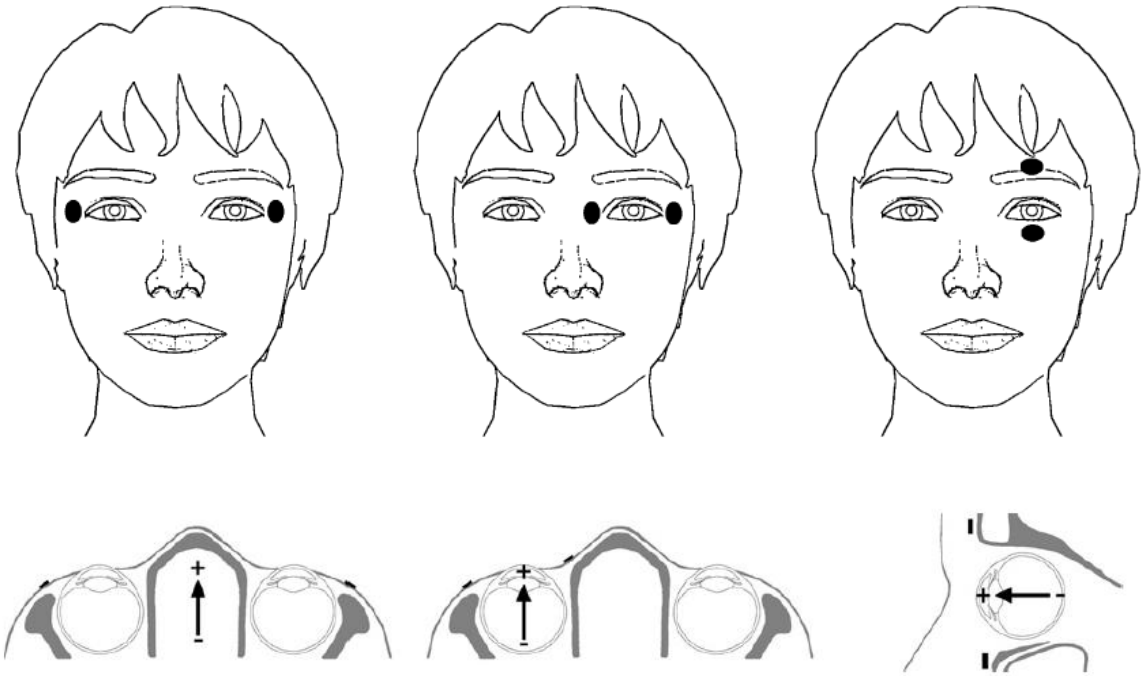


Abbildung 5-2: Platzierung der Elektroden (oben) mit dem resultierenden elektrischen Dipol (unten) **[14]**

Wenn das Auge genau in der Mitte ist messen beide Elektroden das gleiche Potential mit unterschiedlichen Vorzeichen. Ist das resultierende Signal gleich Null. Verschiebt sich jetzt das Feld nach links oder rechts bzw. nach oben nach unten, verschiebt sich somit auch das Elektrische Feld. In Abbildung 2-2 ist dieses Verhalten dargestellt. Das horizontale und vertikale Signal zusammen ergeben einen kartesisches Koordinaten System. Mit Hilfe einer Kalibrierung ist es möglich eine Position in Abhängigkeit vom Mittelpunkt des Koordinatensystems anzugeben. Die Kalibrierung ist notwendig um die Extremwerte zu erkennen.
Entlang der vier Kardinalsbewegungen werden die Augenpositionen definiert: Oben, Unten und Mitte bzw. Rechts, Links. Die „Primäre Position“ wird als Mitte definiert. Als Event ist eine schnelle Bewegung von einer dieser Augenpositionen zu einer anderen definiert. Entscheidend ist die Geschwindigkeit mit der diese Augenbewegung ausgeführt wird, damit nicht langsames Veränderungen wie beispielsweise beim Lesen automatisch zu Events führt. Damit nicht zwischen zwei Zuständen hin und her gesprungen wird überlappen sich die Bereiche. Eine genaue Ausführung ist im Kapitel 6.4 zu finden. Für die Eventerkennung wird daher ein endlicher deterministischer Mealy-Automat verwendet. Da die Event Ausgabe bei der Augenposition Mitte von der vorherigen Position abhängt ist es kein Moore Automat.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

6. Implementierungen

In diesem Kapitel wird die Implementierung der Einzelnen Aufgaben besprochen. Es beginnt mit der Beschreibung des Protokolles für die Steuerung des Signalerfassers. Danach wird die Erweiterung der Mikrocontroller Software behandelt. Mit Hilfe einer bidirektionalen Bluetooth-Verbindung wird das entwickelte Steuerprotokoll umgesetzt. Mit Hilfe der entwickelten Android Software kann der Biosignalerfasser optimal eingestellt werden. Mit den aufgezeichneten Daten wird am Ende des Kapitels die Implementierung des Algorithmus des für das Dipol-Modell für Augenbewegungen erläutert.

6.1 Protokollbeschreibung

Damit der Biosignalerfasser über die Bluetooth-Verbindung gesteuert werden kann, wird ein Klartext- Protokoll entwickelt. Es wird anschließend in der Firmware des Biosignalerfassers sowie in der Android App implementiert. Das Protokoll umfasst alle Konfigurationsregister des Analog Digital Front Ends (ADS1298) und der RTC. Das Datenformat eines Messpaketes soll sich nicht ändern. Das Datenframe der Messwerte des Biosignalerfassers ist in Abbildung 6-1 dargestellt.
Um die Verarbeitung der übertragenen Befehle zu strukturieren, werden Schlüsselwörter definiert, die aussagekräftig und verständlich sind. Diese Schlüsselwörter müssen die Betriebsmodi des Biosignalerfassers abbilden sowie die Einstellungen ermöglichen. Folgende 5 Schlüsselwörter haben sich aus der Analyse ergeben:

Batteriespannung	CH1	CH2	CH3	CH4	CH5	CH6	CH7	CH8	„\r\n“

Abbildung 6-1: Datenframe der Messwerte vom Biosignalerfasser

- „record“

Mit dem Befehl „record“ werden die gemessenen Werte auf die SD Karte geschrieben. Das Datenframe ist in Abbildung 6-1 dargestellt, als Trennzeichen wir ein Semikolon verwendet. Der Biosignalerfasser wird in einen Wartemodus versetzt. Um die Messungen zu starten muss die Bluetooth Verbindung beendet werden, sonst würde bei einer Trennung oder eines Abbruches der Bluetooth-Verbindung die Messung gestoppt werden. Im Betriebsmodus „record“ werden die gemessenen Daten nicht über die Bluetooth-Schnittstelle gesendet.

- „stream“

  Mit dem Befehl „stream“ wird die Messung gestartet und die gemessen Werte werden über Bluetooth an alle verbundenen Geräte gesendet. Sobald das Gerät die Verbindung verliert, stoppt das Device die Messung. Das Datenframe ist das gleiche wie bei der „record“ Funktion und wird in Abbildung 6-1 dargestellt, als Trennzeichen wird ebenfalls Semikolon verwendet. Die Daten werden als ACII- Zeichen übertragen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

ADDRESS	REGISTER	RESET VALUE (Hex)	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
DEVICE SETTINGS (READ-ONLY REGISTERS)										
00h	ID	xx	DEV_ID7	DEV_ID6	DEV_ID5	1	0	DEV_ID2	DEV_ID1	DEV_ID0
GLOBAL SETTINGS ACROSS CHANNELS										
01h	CONFIG1	06	HR	$\overline{\text{DAISY\_EN}}$	CLK_EN	0	0	DR2	DR1	DR0
02h	CONFIG2	40	0	0	WCT_CHOP	INT_TEST	0	TEST_AMP	TEST_FREQ1	TEST_FREQ0
03h	CONFIG3	40	$\overline{\text{PD\_REFBUF}}$	1	VREF_4V	RLD_MEAS	RLDREF_INT	$\overline{\text{PD\_RLD}}$	RLD_LOFF_SENS	RLD_STAT
04h	LOFF	00	COMP_TH2	COMP_TH1	COMP_TH0	VLEAD_OFF_EN	ILEAD_OFF1	ILEAD_OFF0	FLEAD_OFF1	FLEAD_OFF0
CHANNEL-SPECIFIC SETTINGS										
05h	CH1SET	00	PD1	GAIN12	GAIN11	GAIN10	0	MUX12	MUX11	MUX10
06h	CH2SET	00	PD2	GAIN22	GAIN21	GAIN20	0	MUX22	MUX21	MUX20
07h	CH3SET	00	PD3	GAIN32	GAIN31	GAIN30	0	MUX32	MUX31	MUX30
08h	CH4SET	00	PD4	GAIN42	GAIN41	GAIN40	0	MUX42	MUX41	MUX40
09h	CH5SET (1)	00	PD5	GAIN52	GAIN51	GAIN50	0	MUX52	MUX51	MUX50
0Ah	CH6SET (1)	00	PD6	GAIN62	GAIN61	GAIN60	0	MUX62	MUX61	MUX60
0Bh	CH7SET (1)	00	PD7	GAIN72	GAIN71	GAIN70	0	MUX72	MUX71	MUX70
0Ch	CH8SET (1)	00	PD8	GAIN82	GAIN81	GAIN80	0	MUX82	MUX81	MUX80
0Dh	RLD_SENSP (2)	00	RLD8P(1)	RLD7P(1)	RLD6P(1)	RLD5P(1)	RLD4P	RLD3P	RLD2P	RLD1P
0Eh	RLD_SENSN (2)	00	RLD8N(1)	RLD7N(1)	RLD6N(1)	RLD5N(1)	RLD4N	RLD3N	RLD2N	RLD1N
0Fh	LOFF_SENSP (2)	00	LOFF8P	LOFF7P	LOFF6P	LOFF5P	LOFF4P	LOFF3P	LOFF2P	LOFF1P
10h	LOFF_SENSN (2)	00	LOFF8N	LOFF7N	LOFF6N	LOFF5N	LOFF4N	LOFF3N	LOFF2N	LOFF1N
11h	LOFF_FLIP	00	LOFF_FLIP8	LOFF_FLIP7	LOFF_FLIP6	LOFF_FLIP5	LOFF_FLIP4	LOFF_FLIP3	LOFF_FLIP2	LOFF_FLIP1
LEAD-OFF STATUS REGISTERS (READ-ONLY REGISTERS)										
12h	LOFF_STATP	00	IN8P_OFF	IN7P_OFF	IN6P_OFF	IN5P_OFF	IN4P_OFF	IN3P_OFF	IN2P_OFF	IN1P_OFF
13h	LOFF_STATN	00	IN8N_OFF	IN7N_OFF	IN6N_OFF	IN5N_OFF	IN4N_OFF	IN3N_OFF	IN2N_OFF	IN1N_OFF
GPIO AND OTHER REGISTERS										
14h	GPIO	0F	GPIOD4	GPIOD3	GPIOD2	GPIOD1	GPIOC4	GPIOC3	GPIOC2	GPIOC1
15h	PACE	00	0	0	0	PACEE1	PACEE0	PACEO1	PACEO0	$\overline{\text{PD\_PACE}}$
16h	RESP	00	RESP_DEMOD_EN1	RESP_MOD_EN1	1	RESP_PH2	RESP_PH1	RESP_PH0	RESP_CTRL1	RESP_CTRL0
17h	CONFIG4	00	RESP_FREQ2	RESP_FREQ1	RESP_FREQ0	0	SINGLE_SHOT	WCT_TO_RLD	$\overline{\text{PD\_LOFF\_COMP}}$	0
18h	WCT1	00	aVF_CH6	aVL_CH5	aVR_CH7	avR_CH4	$\overline{\text{PD\_WCTA}}$	WCTA2	WCTA1	WCTA0
19h	WCT2	00	$\overline{\text{PD\_WCTC}}$	$\overline{\text{PD\_WCTB}}$	WCTB2	WCTB1	WCTB0	WCTC2	WCTC1	WCTC0

(1) CH5SET and CH6SET are not available for the ADS1294 and ADS1294R. CH7SET and CH8SET registers are not available for the ADS1294, ADS1294R, ADS1296, and ADS1296R.

(2) The RLD_SENSP, PACE_SENSP, LOFF_SENSP, LOFF_SENSN, and LOFF_FLIP registers bits[5:4] are not available for the ADS1294 and ADS1294R. Bits[7:6] are not available for the ADS1294, ADS1296, ADS1294R, and ADS1296R.

Abbildung 6-2: Übersicht der Register des Digital Analog Frontend

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

- „stop“

Der Befehl „stop“ beendet die aktuelle Messung und damit das Senden der Daten über die Bluetooth Schnittstelle. Der Befehl „stop“ ist technisch nicht nötig, da jedes empfange Datenpaket am Mikocontroller die Messung stoppt. Eine ausführliche Erklärung findet sich im Kapitel 6.2.1. Damit das Protokoll intuitiv bleibt, wurde dieser Befehl implementiert.

- „set“

Mit dem „set“ Befehl können die Einstellungen des Biosignalerfassers geändert werden. Der Befehl „set“ hat zwei weitere Teile. Der erste Teil ist der Registernamen des ADS1298, welches gesetzt werden soll. Wenn die RTC gesetzt werden soll, muss „rtc“ angegeben werden. Im letzten Teil des Befehls wird der Wert, welcher gesetzt werden soll, übermittelt. Für das Setzten eines Registers muss eine Integer-Zahl übergeben werden. Für die RTC muss das in Abbildung 6-3 dargestellte Format verwendet werden.

- „get“

  Mit dem „get“ Befehl kann der aktuelle Wert eines Registers oder der RTC ausgelesen werden. Mit dem Registernamen oder „rtc“ kann ausgewählt werden, welcher Wert übermittelt wird.

  Als Antwort sendet der Biosignalerfasser den gefragten Wert.

Die Übertragenen Befehle werden vom Mikrocontroller ausgewertet und die zugehörigen Funktionen ausgeführt. So werden „get“ oder „set“ Befehle für die RTC übersetzt und mit Hilfe eines I²C Bus an die RTC kommuniziert. Beim Analog Digital Front End werden die „get“ oder „set“ Befehle übersetzt und über einen UART weiter kommuniziert.

6.1.1 Analog Digital Frontend Register

In Abbildung 6-2 sind alle Register des ADS1298 abgebildet. Mit den Registern CH1SET bis CH8SET können individuelle Einstellungen für jeden Kanal getroffen werden. Für jeden Kanal kann die Verstärkung separat eingestellt werden. Die Detail-Beschreibung aller Register kann im Datenblatt für den ADS 1298 von TI nachgelesen werden [42]. Alle Register können mit dem „set“ Befehl gesetzt oder mit dem „get“ Befehl ausgelesen werden.

6.1.2 Real Time Clock

Die RTC kann mit der Option „rtc“ neu gesetzt werden. Das Datum hat im folgenden Format eingegeben zu werden: „(JJ)JJ;MM;DD;N;hh;mm;ss“. Das Jahr kann wahlweise 4 stellig oder zweistellig angegeben werden. Für N muss der Wochentag eingetragen werden, wobei 1 für Sonntag steht, 2 für Montag usw., bis 7 für Samstag. Die Jahreszahl muss zwischen 0 und 99 für 2-stellige Angaben liegen, für 4-stellige muss sie zwischen 2000 und 2099 liegen. Eine Beschreibung der weiteren Abkürzungen kann in Tabelle 6-1 gefunden werden. Das übergebene Datum und die Uhrzeit werden auf Plausibilität geprüft. Falls die Daten nicht stimmen, wird eine Fehlermeldung für den ersten erkannten Fehler ausgegeben.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

JJJJ	MM	DD	N	hh	mm	ss

Abbildung 6-3: Datenformat des Timestamp der RTC beim „set“ Befehl

Abkürzung	Beschreibung	Wertebreich
hh	Stunde im 24h Format	0-24
mm	Minuten	0-59
ss	Sekunden	0-59
JJJJ	Jahreszahl	2000-2099
MM	Monat	1-12
DD	Tag	1-31
N	Wochentag mit Sonntag beginnend	1-7

Tabelle 6-1: Beschreibung des RTC Timestamp

6.1.3 Datenübertragungsrate

Die benötigte Datenrate lässt sich durch Anzahl der übertragenen Zeichen mal benötigte Bits pro Zeichen mal Abtastrate [43] berechnen. In Abbildung 6-1 ist das Datenframe genau abgebildet. Jedes Zeichen wird im American Standard Code for Information Interchange (ASCII) codiert. Das reine Nutzsignal besteht aus 8 Kanälen mit jeweils max. 24 Bit. Ein Weiterer 24 Bit Kanal kommt noch für die Batteriespannung am Beginn des Datenframes hinzu. Der Messbereich für einen 24 bit Kanal geht von -8388607 bis 83886080. Daher braucht ein Kanal 7 ASCII Zeichen plus Vorzeichen. Nach jedem Kanal kommt noch ein Trennzeichen dazu. Das Datenpaket wird mit zwei Zeichen „\n\r“ abgeschlossen. Zusammen hat daher ein komplettes Datenpaket max. 83 Zeichen. Im ASCII Code ist dies eine 7-Bit-Zeichencodierung [43], damit werden insgesamt 581 Bit pro Datenframe übertragen. Da die Abtastrate 250 Hz beträgt, ergibt sich eine benötige Übertragungsrate von max. 145,250 kbit pro Sekunde. Das SPP Protokoll hat eine maximale Datenübertragungsrate von 250 kbit pro Sekunde. Damit wird die benötigte Datenübertragungsrate von SPP Protokoll unterstützt.

Da bei einer Bluetooth-Verbindung die unteren Schichten verantwortlich sind für die sichere und korrekte Übertragung der Daten ist es in diesem Protokoll nicht nötig sich jeden erfolgreich ausgeführten Befehl mit einer Erfolgsmeldung bestätigen zu lassen. Tritt ein Fehler bei der Bearbeitung eines Befehls auf wird dieser an den die Gegenstelle übermittelt. Wird ein ungültiger Befehl übermittelt wir als Antwort „unknown Command“ zurückgesendet. Sollte es nötig sein zu überprüfen wie ein Register gesetzt ist kann dies jederzeit mit einem „get“ Befehl abgefragt werden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

6.2 Mikrocontroller Software

Die Mikrocontroller Firmware ist eine Weiterentwicklung aus der bereits bestehenden Software. In Abbildung 6-4 ist der Funktionsablauf der bestehenden Software dargestellt. Nach dem Start der Software werden alle Initialisierungen vorgenommen. Ob die Messung läuft oder nicht wird mit Hilfe der Taste auf dem Messverstärker bestimmt. Wenn diese Taste gedrückt wird, wird ein Interrupt ausgelöst, welcher die Variable für die Messung setzt. Die zwei Hauptfunktionen „aktuelle Daten auf SD Karte schreiben“ und „Messwerte über Bluetooth-Modul senden“ bleiben unverändert der Ablauf in der Main-Funktion ändert sich aber grundlegend. Welche Funktion ausgewählt wird, hängt davon ab, ob im SD-Kartenslot des HW-Devices eine SD-Karte erkannt wird oder nicht. Dadurch, dass mit der Weiterentwicklung eine bidirektionale Verbindung mit dem Biosignalerfasser möglich ist, kommen neue Abläufe und Funktionalitäten hinzu. Der Funktionsablauf der neuen Software ist in Abbildung 6-5 dargestellt. Die Funktionen, die rot hinterlegt sind, wurden dabei aus der bereits bestehenden Software übernommen, ohne eine Änderung. Der Ablauf und die anderen Funktionsblöcke sind im Rahmen dieser Arbeit entwickelt worden.

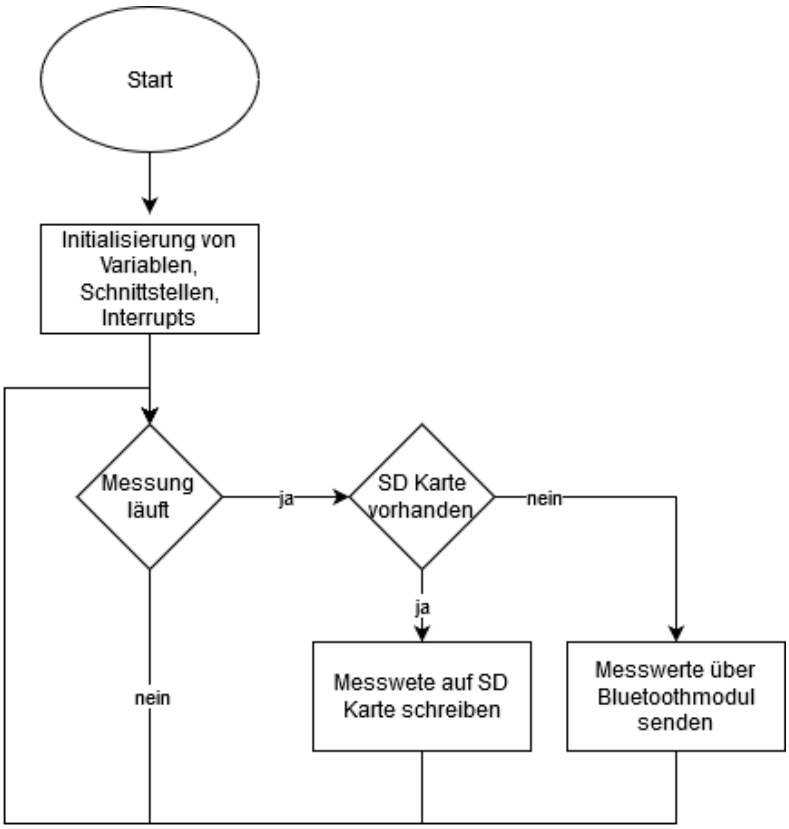


Abbildung 6-4: Funktionsablauf übernommener Mikrocontroller Software

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

6.2.1 Funktionsablauf der Firmware

Der neue Funktionsablauf der Firmware ist in Abbildung 6-5 dargestellt. Mit Hilfe von zwei Statusvariabelen wird der aktuelle Betriebsmodus des Biosignalerfassers gespeichert. Die erste Steuervariable speichert den Zustand der Bluetooth-Verbindung. Sie wird nur durch den Auswerteblock des Übertragungsprotokolls geändert. Die zweite Statusvariablen („Status“) kann entweder durch einen Interrupt neu gesetzt werden, zB. der durch die Taste am Biosignalerfasser ausgelöst wird, oder durch

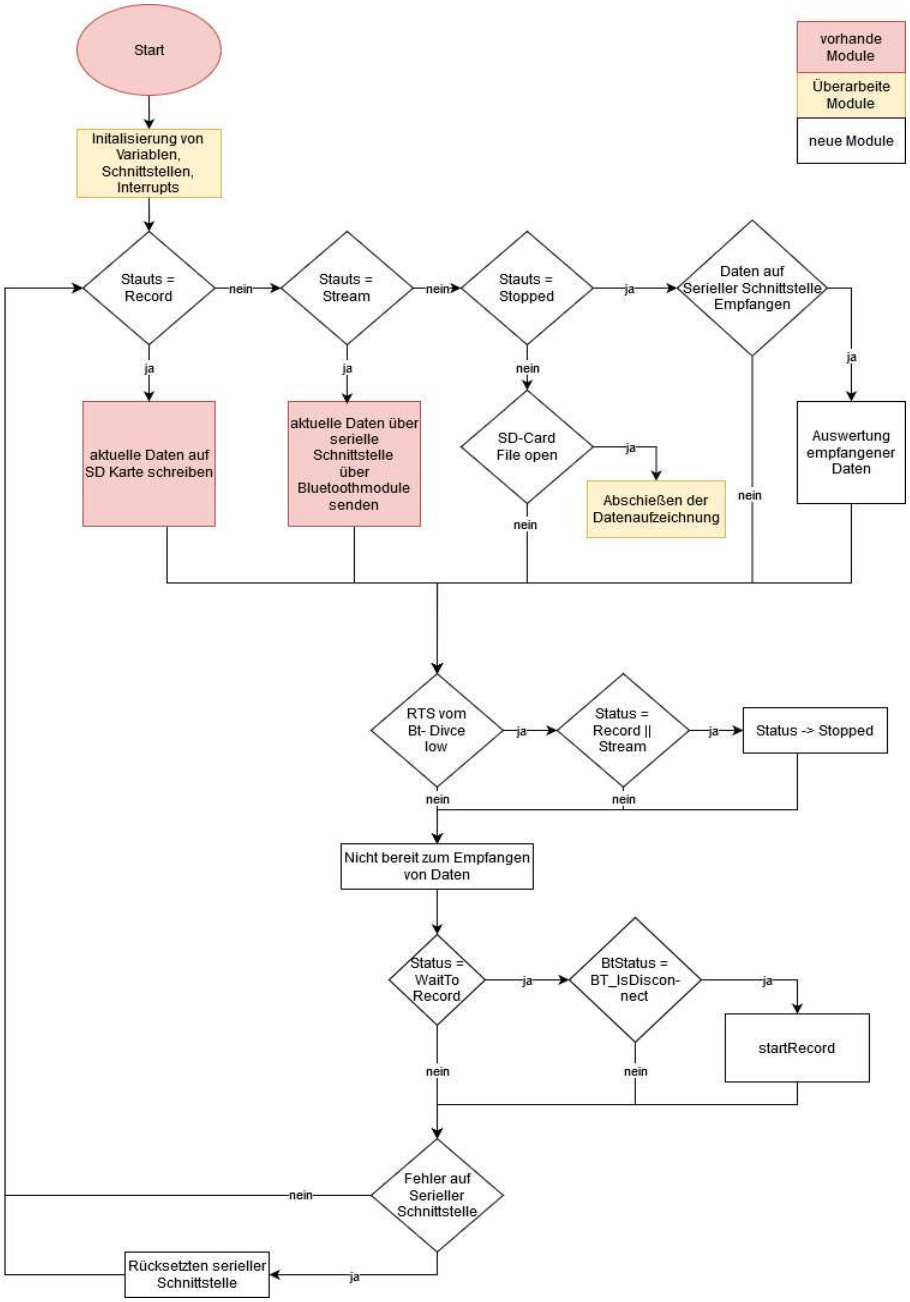


Abbildung 6-5: Funktionsablauf überarbeitete Mikrocontroller Software

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

den Auswertungsblock des Übertragungsprotokolls. Diese Statusvariable kann folgende Zustände haben:

- Record: dabei werden die Messwerte auf die SD-Karte geschrieben
- Stream: dabei werden die Messwerte über die Bluetooth-Schnittstelle gestreamt
- Stopped: der Biosignalerfasser misst nicht und wartet auf eine Eingabe
- WaitToRecord: der Biosignalerfasser hat die Aufgabe bekommen, in den Record Modus zu wechseln, ist aber noch mit einem Bluetoothgerät verbunden. Sobald die Bluetoothverbindung getrennt ist, kann mit der Datenaufzeichnung begonnen werden.

Der WaitToRecord-Modus ist notwendig, da das Schreiben von Daten auf die SD Karte in einem kontinuierlichen Bitstrom erfolgen muss. Andernfalls kann es im SD-Kartenmodul zu einem Fehler kommen, der zu einem korrupten File führt. Dieser Bitstrom kann durch einen Interrupt oder durch das Verarbeiten von empfangenen von Bluetooth- Daten auf der seriellen Schnittstelle abreißen und somit das File auf der SD-Karte unbrauchbar machen. Damit der Bitstrom nicht abreißen kann, werden beim Auftreten eines Interrupts die aktuellen Messwerte fertig auf die SD-Karte geschrieben, das File abgeschlossen und dann erst der Interrupt fertig behandelt. Daher wird in der Firmware kein Interrupt zum Abarbeiten der seriellen Schnittstelle verwendet. Es wird nach der Abarbeitung der drei Grundfunktionen überprüft, ob das Bluetooth-Device Daten senden möchte, indem die Ready to Send (RTS) Leitung abgefragt wird. Wenn die RTS auf low ist wird die Statusvariable „Status“ auf „Stopped“ gesetzt um das File abzuschließen, sofern der Betriebsmuodus Record oder Stream aktiv ist. In der nächsten Interation wird dann der Funktionsblock „Auswertung empfangener Daten“ aufgerufen und die Daten des Bluetooth Modules werden empfangen und ausgewertet.
Falls der aktuelle Status „WaitToRecord“ ist, wird überprüft, ob das Bluetoothgerät noch verbunden ist. Solange das Bluetoothmodul eine Verbindung hat, wird nicht auf die SD-Karte geschrieben. Grund dafür ist, dass eine Nachricht an den Mikrocontroller gesendet wird, sobald der Status der bluetoothverbindung sich ändert. Diese Nachricht bewirkt, dass die Record-Funktion umgehend beendet wird. Erst wenn keine Bluetooth-Verbindung besteht, wird mit der Datenaufzeichnung auf die interne SD-Karte begonnen. Als letztes wird überprüft, ob die serielle Schnittstelle einen Fehler erkannt hat. Sollte dies der Fall sein wird die Schnittstelle zurückgesetzt.

6.2.2 Implementierung des Übertragungsprotokolls

Ein wesentlicher Teil der Arbeit ist, das entwickelte Übertragungsprotokoll zu implementieren. Im Funktionsblock „Auswertung empfangender Daten“ befindet sich der Empfangsteil der Implementierung des in Kapitel 6.1 beschrieben Übertragungsprotokolls, sowie der Übertragungsteil für ausgehende Nachrichten, die im Zuge der bidirektionalen Kommunikation entstehen. Das Versenden von Mess-Datenpaketen wird nicht in der Auswerte Funktion behandelt.
Damit die eine bei der bidirektionalen Kommunikation die „request to send“ (RTS) und clear to send (CTS) Signale zwischen den Mikrocontroller und dem Bluetooth-Module verwendet werden könnten wurde die Hardware am Biosignalerfasser geändert. Konkret wurden an der Platine des Bluetooth-Moduls zwei Widerstände eingelötet. Es handelt sich um die Widerstände R3 und R5 im Schaltplan. Die Widerstände R4 und R6 wurden ausgelötet. Daher besteht keine Verbindung zwischen den 7 und 11 Pins des Steckers mit Ground. Ein Ausschnitt des Schaltplans ist in Abbildung 6-6 zu sehen. Der

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

komplette Schaltplan des Bluetooth-Moduls vom Biosignalerfassers ist im Anhang zu finden. Wenn Daten am Bluetooth-Empfänger empfangen werden, versucht das Bluetooth-Device diese Daten über den UART zum Mikrocontroller zu senden. Sobald das Bluetooth-Modul die RTS Leitung auf low legt, stoppt der Mikrocontroller, nachdem er den Status der Leitung abgefragt hat, den aktuellen Modus und ruft die Funktion „Auswertung empfangener Daten“ auf. Der Funktionsablauf ist Abbildung 6-7 dargestellt. Als erstes werden so lange Daten am UART empfangen, bis das Bluetooth-Device mit Hilfe der RTS- Leitung dem Mikrocontroller signalisiert, dass alle Daten gesendet wurden. Da das Bluetooth-Device Statusmeldungen über den UART verschickt, müssen diese Nachrichten ebenfalls

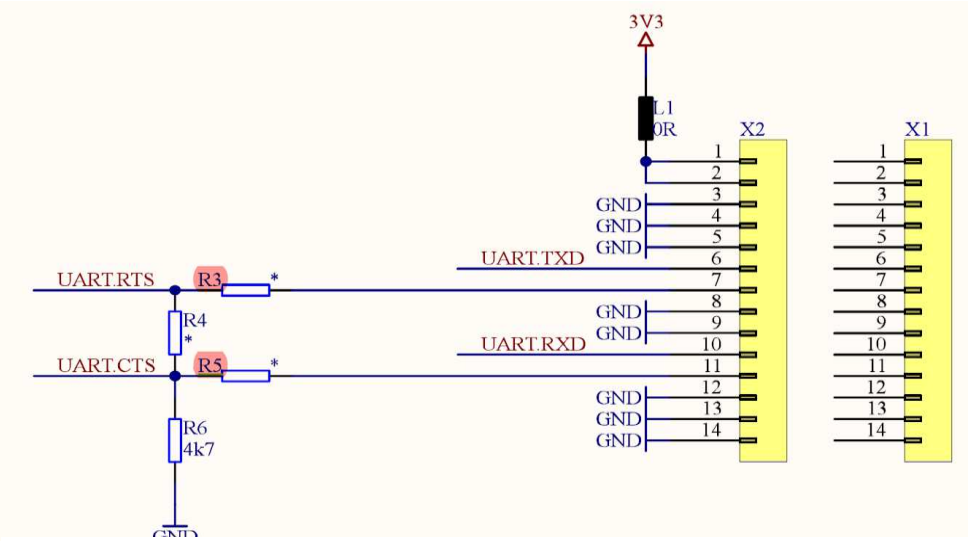


Abbildung 6-6: Schaltplanteil Bluetooth-Platine Biosignalerfassers.

über den Block „Auswertung empfangener Daten“ ausgewertet werden. Als erstes wird überprüft ob die Daten korrekt mit „\n“ abgeschlossen sind. Danach wird festgestellt, ob es sich um Nutzdaten oder um eine Statusnachricht des Bluetooth-Modules handelt. Wenn das vorletzte Zeichen der empfangenen Zeichenkette „0x03“, ist handelt es sich um ein Datenpaket vom Bluetooth-Device, da alle Nachrichten mit „0x03 \n“ abgeschlossen werden.

Wenn die empfangen Daten Befehle aus dem Übertragungsprotokoll sind, werden sie beim ersten Leerzeichen getrennt, wenn sich ein Leerzeichen in der Zeichenkette befindet. Dadurch wird nur der erste Teil des Befehles in eine Zeichenkette gespeichert. Da unter C die „switch“ Funktion nicht für Zeichenketten zu Verfügung steht, wird mit if-else Verzweigungen gearbeitet. Dadurch kann zwischen den vier Hauptbefehlen unterschieden werden. Je nach Befehl wird wieder eine eigne Funktion aufgerufen. Bei „get“ oder „set“ Befehlen wird abhängig vom zweiten Datenblock der empfangen Nachricht, der nächsten Teil auswertet werden. Auch diese Auswertung erfolgt wieder mittels if-else Verzweigungen. Dadurch wird das passende Register ermittelt. Im letzten Teil der Nachricht, wenn es sich um einen „set“ Befehl handelt, wird der zu setzende Wert empfangen. Danach kann das Register gesetzt werden. Im Falle eines „get“ Befehles wird nach der Ermittlung des Registers dieses ausgelesen und eine Nachricht mit dem Inhalt des Registers zurückgesendet. Diese Nachricht wird über den UART an das Bluetooth-Modul gesendet und von dort weiter über die Bluetooth-Verbindung. Falls ein Befehl nicht ausgewertet werden kann, wird eine Fehlermeldung gesendet. Die anderen beiden

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Befehle starten die Straem oder Record Funktion des Mikrocontrollers. Sollte ein Fehler bei der Decodierung oder beim Starten einer Funkion aufgetreten sein, wird diese Fehlermeldung über den UART an das Bluetooth- Device geschickt und so an den Absender übermittelt.
Daten, die vom Bluetooth-Device über den UART empfangen werden, bestehen immer aus vier Zeichen, wobei die letzten beiden Zeichen immer den Abschluss der Nachricht mit „0x03“ und „\n“ markiert. Danach wird überprüft, ob es sich um eine Nachricht „Link Established“ handelt oder um die Nachricht „Link Released“. Link Established ist mit den Zeichen 0x105 und 0x12 codiert. Link Released ist mit den Zeichen 0x105 und 0x14 codiert. Je nach Nachricht wird das Statusflag für die Bluetooth-Verbindung gesetzt.

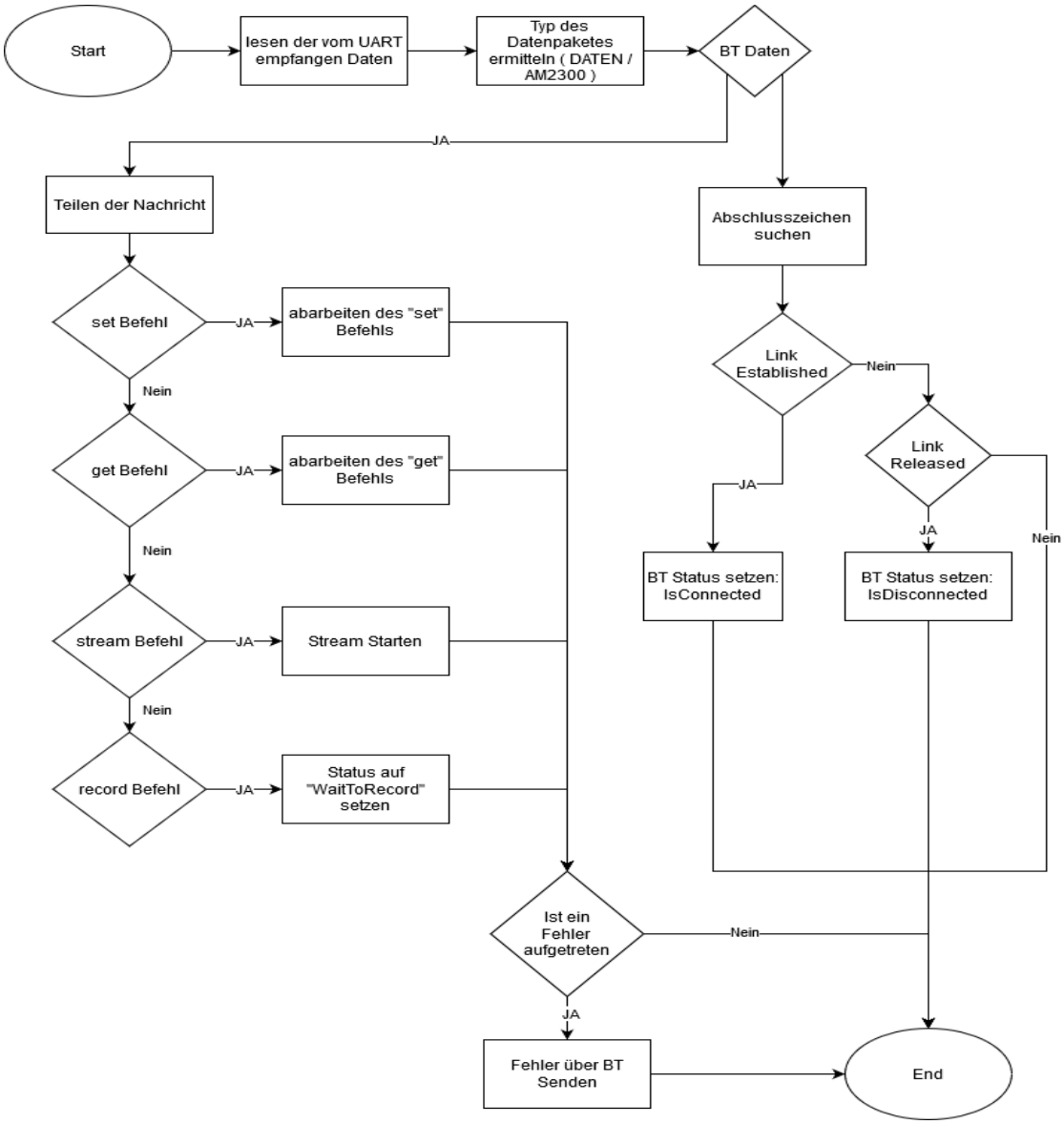


Abbildung 6-7 Funktionsablauf Auswertung empfangener Daten

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

6.3 Bestandteile der Android® Applikation

Die Android App soll sich einfach und schnell mit dem Signalerfasser verbinden können und die einzelnen Kanäle zeitnah darstellen. Als Grundlage für die Entwicklung von Android Programmen wurde das Buch „Android 4 Apps entwickeln mit dem Android SDK“ [22] verwendet. Es wurden unterschiedliche Programme und Testprogramme im Laufe dieser Arbeit entwickelt. Im Folgenden werden die wichtigsten Funktionsblöcke des fertigen Programmes beschrieben. Damit sich die Software mit dem Biosignalerfasser verbinden kann, werden die Methoden, die bereits in Android implementiert sind, verwendet. Dies gilt sowohl für die Suche nach Bluetooth-Geräten, als auch für den eigentlichen Verbindungsaufbau. Um die Parameter des Biosignalerfassers setzen zu können, werden die Methoden, die Android für Einstellung von Applikationen bereithält, verwendet. Mit Hilfe einer externen Library werden schlussendlich die Daten in verschiedenen dynamischen Messkurven dargestellt. Das im Kapitel 0 beschriebene Protokoll ist in der App ebenso implementiert worden.

6.3.1 Bluetooth Verbindung in Android®

Wie mit Android eine Bluetooth-Verbindung aufgebaut werden kann, wird komplett in einem Beispiel in der Android-Dokumentation gezeigt. Dabei wird eine Bluetooth-Verbindung über das SPP aufgebaut und zum Austausch von Chat-Nachrichten verwendet [44]. Beispiel Code wie in Android eine Verbindung aufgebaut werden kann ist in Abbildung 6-8 kurz dargestellt. Als Voraussetzung um Bluetooth in der Applikation verwenden zu können, muss in der Datei manifest.xml, die Erlaubnis für BLUETOOTH und BLUETOOTH_ADMIN gesetzt sein. Damit bekommt die Applikation die Berechtigung auf das Bluetooth-Gerät zuzugreifen. Jede Verbindung beinhaltet einen sogenannte Universally Unique Identifier (UUID). Die UUID ist in ISO/IEC 11578 definiert [45]. Im Wesentlichen besteht eine UUID aus einer 128-Bit-Zahl, diese dient zur Identifikation. Im Bluetooth-Standard [34]

```
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;

private BluetoothAdapter mBtAdapter= null;

//check if Bluetooth Adapter available
mBtAdapter= BluetoothAdapter.getDefaultAdapter();
if (mBtAdapter == null)  //if not available show message
{ Toast.makeText(this, "There is no Bluetooth device availa-
ble", Toast.LENGTH_LONG).show();
finish();
return;
}
//turn on Bluetooth Adapter if is not enabled
if(!mBtAdapter.isEnabled())
{  Intent enableIntent = new Intent(BluetoothAdapter.AC-
TION_REQUEST_ENABLE);
   startActivityForResult(enableIntent, REQUEST_ENA-
BLE_BT);
}
```

```
//connect to device with address (example address)
private String address="00:18:DA:01:AA:DE";

private static final UUID MY_UUID = UUID.from-
String("00001101-0000-1000-8000-00805F9B34FB");

// Get a BluetoothSocket from given BluetoothDevice
mbtDevice = mBtAdapter.getRemoteDevice(address);

BluetoothSocket mBtSocket = mbtDevice.createRfcommSock-
etToServiceRecord(MY_UUID);
mBtSocket.connect();
InputStream mBtInStream = mBtSocket.getInputStream();
OutputStream mBtOutStream = mBtSocket.getOutputStream();
```

Abbildung 6-8: Code Stücke zum Erstellen einer Bluetooth Verbindung

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

wird beschrieben, wie die UUID aufgebaut sein sollte. Durch einen korrekten Aufbau der UUID kann der SPD-Dienst erkennen, welches Service vom Gerät unterstützt wird. In dieser Arbeit besteht daher der erste Block der UUID mit dem Code für das SPP mit dem Wert 0x1101. Damit ist sichergestellt, dass jedes Gerät erkennt, dass der Biosignalerfasser eine SPP Verbindung unterstützt. Mit Hilfe der Klasse „DeviceListActivity.java“ wird ein Dialog gestartet, bei dem das Bluetooth- Gerät ausgewählt werden kann. Der Dialog startet den Standard-Dialog, welcher in den Android- Geräteeinstellungen für die Suche und Auswahl eines Gerätes zuständig ist. Das gewählte Bluetooth- Gerät wird in den Einstellungen der App hinterlegt. Damit kann die App auf Wunsch des Benutzers eine Verbindung aufbauen. Wenn eine Verbindung erfolgreich hergestellt ist, wird ein eigener Thread im Hintergrund erstellt. Dieser übernimmt dann das Management der Daten, die empfangen oder gesendet werden sollen. Sobald eine Verbindung zustande kommt, werden die Einstellungen abgerufen und in der App hinterlegt. Sobald eine Verbindung zwischen dem Biosignalerfasser und der Android-App aufgebaut ist, wird mit Hilfe des set-Befehls die RTC im Biosignalerfasser neu gesetzt.

6.3.2 Dynamische Messkurven in Android® darstellen

Um Messkurven in Echtzeit darstellen zu können, wird eine externe Library verwendet. In dieser Arbeit wurde „AndroidPlot“ in der Version 0.6.0 verwendet [46]. Diese Library wird schon in anderen Arbeiten [47], [48] verwendet um Messkurven dynamisch darzustellen. Eine ausführliche Dokumentation kann auf der Webseite des Projektes gefunden werden [46]. In dieser Anwendung wird die Funktion des „XYPlot“ verwendet. Der erzeugte Graph bietet die Möglichkeiten von Beschriftungen der Achsen, sowie das Setzen von fixen Grenzen der Achsen. Auch diverse Parameter für die Darstellung wie Farbe, Strichstärken und vieles mehr kann eingestellt werden. „Android Plot“ verfügt über zwei Möglichkeiten, wie ein Graph aktualisiert wird. Die einfache Möglichkeit ist, nach jeder Mani-

```
//number of points to plot in history
private static final int HISTORY_SIZE = 250;
private XYPlot channelHistoryPlot = null;
            private SimpleXYSeries channelHistorySeriesSimple
=null;
            private klavigXYListplot channelHistoryY=null;
            private SimpleXYSeries channelHistoryXY;
//Androidplot

channelHistoryPlot = (XYPlot) myView.findView-
ById(R.id.myYTimePlot);
channelHistoryPlot.setRangeBoundaries(-500000, 500000,
Boundary
        Mode.AUTO);
channelHistoryPlot.setDomainBoundaries(0,HIS-
TORY_SIZE,Boundary
        Mode.FIXED);
channelHistoryPlot.setDomainStepValue(1);
channelHistoryPlot.setTicksPerRangeLabel(1);
```

```
channelHistoryPlot.setDomainLabel("Index");
channelHistoryPlot.setRangeLabel("ADC Channel Value");
channelHistoryPlot.getGraphWidget().setPaddingLeft(30);
channelHistoryPlot.getGraphWidget().setPaddingTop(20);
channelHistoryPlot.getGraphWidget().setPaddingBottom(20);
hannelHistoryY = new klavigXYListplot("Data", HIS-
TORY_SIZE);
Paint lineFill =new Paint();
lineFill.setAlpha(200);
LineAndPointFormatter myformatter = new LineAndPointFor-
matter(Color.BLUE, Color.BLACK, null, null);
channelHistoryPlot.addSeries (channelHistoryY, myformatter)
channelHistoryPlot.redraw();

private void updatePlot(String data) {
channels[ChannelView]=Long.parseLong(channelsSt[Chan-
nelView]);
channelHistoryY.addY(channels[ChannelView]);
channelHistoryPlot.redraw();}
```

Abbildung 6-9: Code Stück für einen XY Plot wobei auf X-Achse Zeit und y-Achse Wert dargestellt werden

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

pulation der Daten die Grafik neu zeichnen zu lassen. Bei hohen Abtastraten kann dies zu Verzögerungen führen. Daher wurde in dieser Arbeit die zweite Methode verwendet. Dabei wird im Hintergrund ein eigener Thraed, der kontinuierlich die Grafiken aktualisiert, gestartet.
Viele Anzeigen im technischen Bereichen sind Anzeigen bei denen der Signalverlauf von links nach rechts dargestellt wird, sobald das Signal den rechten Rand der Anzeige erreicht hat läuft das Signal von rechts nach links und dann aus der Anzeige heraus. Im medizinischen Bereich hat sich eingebürgert, dass das Signal ebenfalls von links nach rechts läuft, sobald das Signal den rechten Rand der Anzeige erreicht hat beginnt das Signal wieder auf der linken Seite der Anzeige und läuft dann wieder auf die rechte Seite. Im technischen Bereich werden solche Anzeigen z.B.: bei Oszilloskopen verwendet. Diese Art der Anzeige ist historisch bedingt durch die ersten Anzeigen die mittels Elektronstrahl erzeugt wurden.
In dieser Arbeit wurde die Klasse „SimpleXYSeries“ verwendet, um die dynamische Messkurve darzustellen. Im Objekt dieser Klasse werden die Werte, die für die Erstellung des Graphen benötigt werden, verwaltet. So verfügt die Klasse über eine Methode, die ein Element zu X oder Y Koordinaten hinzufügt, und eine Methode, die ein Element entfernt. Damit die Anzeige nach den oben beschreiben Schema das Signal darstellt muss immer ein Index mitlaufen damit der aktuelle Messpunkt an die richtige Stelle auf der Y-Achse ersetzt wird. Dies wird von der Funktion „addY“ implementiert. Da alle Messpunkte mit einer Linie verbunden werden, wirkt es oft in Screenshots als ob ein Sprung im Signal ist. Für die Grafik, die den Messwert über die Zeit darstellt, wird nur ein Y-Wert benötigt. Auf der X-Achse werden kontinuierlich die letzten 250 Messwerte angezeigt. Mit einer Abtastrate von 250Hz würde damit nur die letzte Sekunde darstellbar sein. Damit eine längere Darstellung möglich ist, wird nur jeder fünfte Wert oder zehnte Wert in dem Plot angezeigt. Die Einstellung kann in den Settings vorgenommen werden. Mehr Messpunkte auf der X-Achse sind auf den kleinen Displays mit begrenzter Auflösung nicht sinnvoll, da sich sonst die Messpunkte überlappen könnten. Für den XYPlot wird ein Kanal den X-Werten und ein Kanal den Y Werten zugeordnet. Auch bei dieser Grafik wird nur jeder fünfte oder zehnte Messpunkt verwendet. In Abbildung 6-9 werden Code Stücke gezeigt für einen Simplen XY Plot.

6.3.3 Fragments in Android®

Fragmente wurden bei Android eingeführt, da die Displaygrößen mit Tablets sehr unterschiedlich geworden sind. In einer Activity wird jeweils ein Dialog nach dem anderen sequenziell abgearbeitet. Fragmente sind ein Teil der Activity und laufen unabhängig voneinander parallel ab. Der Life Cycle eines Fragments ist sehr eng mit dem der Activity verbunden. Fragmente besitzen jedoch noch zusätzliche Callback-Methoden, da sie innerhalb einer Activity angezeigt und ausgeblendet werden können. Da ein ausgeblendetes Fragment nicht zerstört wird, laufen im Hintergrund noch alle Routinen weiter.

In der App wurden drei Fragmente für die Funktion der App implementiert und 1 Fragment für die Einstellungen. Beim ersten Fragment wird ein beliebiger Kanal über die Zeit dargestellt. Das Fragment wurde im Code mit „Fragment1y“ benannt. Im zweiten Fragment werden zwei Graphen angezeigt, damit es möglich ist, zwei verschiedene Kanäle gleichzeitig darzustellen. Im Falle eines EGO Signals könnte das ein horizontaler und ein vertikaler Kanal sein. Dieses Fragment wurde mit „Fragment2y“

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

benannt. Im Wesentlichen wurden in diesem Fragment lediglich zwei Fragmente vom Typ „Fragment1y“ eingebaut. Im dritten Fragment wird ein XY Plot von zwei Kanälen erzeugt. Im Code kann dieses Fragment durch seinen Namen“ FragmentXY“ identifiziert werden. Der Benutzer kann den Kanal für die X Werte und die Y Werte auswählen. Welches Fragment dargestellt wird, kann vom Nutzer ausgewählt werden. Dadurch, dass die Fragmente beim Ausblenden nicht zerstört werden, können im Hintergrund die nötigen Funktionen weiter laufen um die Datenbasis der Graphen aktuell zu halten. Das gewährleistet ein übergangsloses Ein- und Ausblenden der verschiedenen Anzeigen. Das letzte implementierte Fragment dient dazu, die Einstellungen der App anzuzeigen. Die Klasse „KlavigControlFragment“ managt den Wechsel zwischen den Fragmenten.

6.3.4 Einstellungen der Applikation

Android bietet eine ganze Reihe von Klassen für das Setzen von Einstellungen an. In jedem Layout ist eine Schaltfläche optional verfügbar, um in die Einstellungen der App zu kommen. Dabei können die Einstellungen wie ein normaler Dialog aufgebaut werden oder auch als Fragment. Jede Einstellung kann mit verschieden Elementen wie Button, Schieberegler, Auswahlmenü etc. erstellt werden. Jede dieser Funktionen bietet eine Callback Funktion.
Um das Einstellen der Verstärkungsparameter einfacher zu gestalten, gibt es auch die Möglichkeit, die Kanäle 1-3 und 4-8 gemeinsam zu verstellen. Hintergrund hierfür ist, dass die Kanäle 1-3 meist für EOG Signale und die Kanäle 4-8 für ein EEG eingesetzt werden. Sobald eine Einstellung geändert wird, wird diese Änderung an den Biosignalerfasser übertragen. Eine Änderung der Einstellungen ohne einer Verbindung zum Biosignalerfasser ist nicht möglich.
Zusätzlich zu den Optionen des Biosignalerfassers kann noch angegeben werden, ob die empfangenen Daten in ein File geschrieben werden. Der Filenamen kann beliebig gewählt werden, optional kann der Timestamp, beim Starten der Messung, verwendet oder an den Filenamen angehängt werden. Damit Android auf den Speicher mit Schreibrechten zugreifen kann muss die zugehörige Permission im Android Manifest hinterlegt sein. Die genau Permission heißt „WRITE_EXTERBAL_STORAGE“.

6.3.5 Datenverarbeitung in der Applikation

In der Android-Applikation sind die Möglichkeiten der Datenverarbeitung begrenzt, da noch keine portable Runtime vom Matlab für Andoid zu Verfügung steht. Es ist jedoch möglich einen C-Code aus Matlab zu generieren der dann in Android eingebaut werden kann. Beim Empfang eines Datenpakets wird auf Vollständigkeit des Datenframes überprüft. Der erste Verarbeitungsschritt ist die Überprüfung ob es sich um ein Datenpaket oder um ein Einstellungsnachricht handelt.
Bei einem Datenpaket wird überprüft, ob das Paket in 9 Kanäle getrennt, mit einem Semikolon, zerlegt werden kann und ob jeder Kanal aus Zahlwerten besteht. Eine Weiterverarbeitung der Daten mit Hilfe von Matlab ist derzeit noch nicht implementiert, daher wird ein korrektes Datenpaket, den einzelnen Fragmenten zur Bearbeitung weitergereicht. In den Fragmenten werden die einzelnen Werte der Kanäle zu den zugehörigen Variablen vom Type „SimpleXYSeries“ hinzugefügt. Danach wird der älteste Wert aus der Variable entfernt. Sollte die Option, dass Daten aufgezeichnet werden, gewählt sein, werden in der Hauptaktivität des Programmes die Daten an das zugehörige File angefügt. Bei einem

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Einstellungspakt wird, im Falle einer korrekten Nachricht, die Einstellung in den Android Settings dem geänderten Status des Bisosignalerfassers angepasst.

6.4 Realisierung in MATLAB®

In Kapitel 0 wurde das Dipol-Modell für Augenbewegungen ausführlich beschrieben. Entscheidend ist, dass die Signale unabhängig voneinander betrachtet werden können. Die Implementierung in MATLAB® wurde in drei Programmteile unterteilt. Der modulare Aufbau des Programmes erleichtert das Testen und die Wiederverwendbarkeit des Codes für die einzelnen Projekte. Einstiegspunkt in die Software bietet die Funktion „EOG_online“, deren Funktionsablauf in Abbildung 6-10 dargestellt ist.

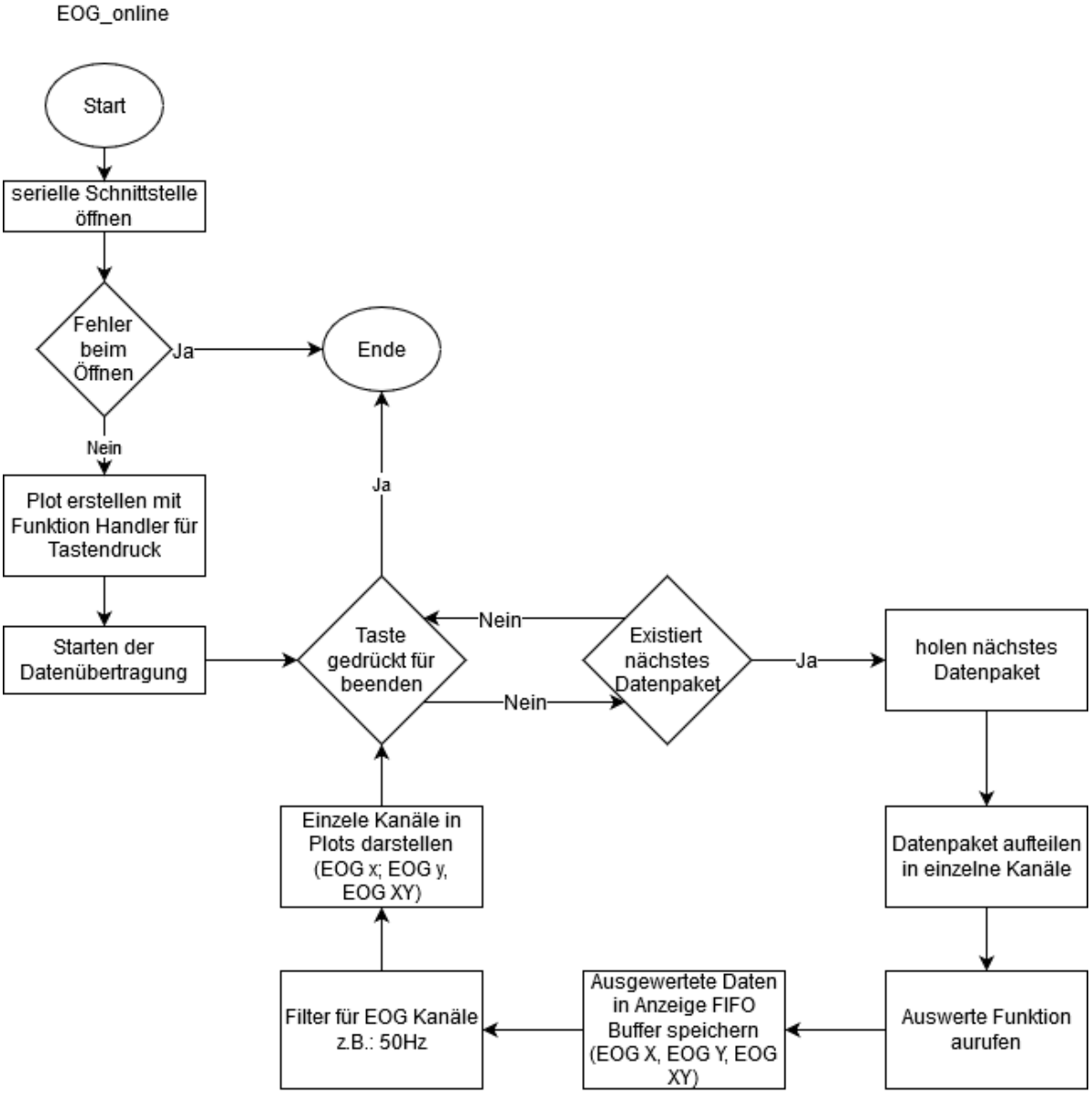


Abbildung 6-10 Funktionsablauf von EOG_online

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die Funktion stellt eine Testumgebung zu Verfügung um den Algorithmus für das EOG Signal testen zu können. Es wird zum einem die Verbindung zur seriellen Schnittstelle des Biosignalerfasser aufgebaut, und zum anderen die Anzeigen und Eingaben für den Benutzer oder die Benutzerin bereitgestellt. Um die Signale darstellen zu können, werden drei Funktionsgraphen erzeugt. Davon zeigen zwei den Verlauf des vertikalen und des horizontalen EOG Signales über die Zeit dar. Der dritte Graph wird für einen XY Graphen verwendet. Mit dem Drücken einer beliebigen Taste wird das Programm beendet. Im Funktionsblock „Auswertefunktion aufrufen“ wird die Signalverarbeitung und der Algorithmus für die Eventerkennung durchgeführt. Nach der Auswertung werden die Werte in einem „First in first out“ Buffer (FIFO-Buffer) gespeichert, um den Verlauf der Werte darstellen zu können. Als

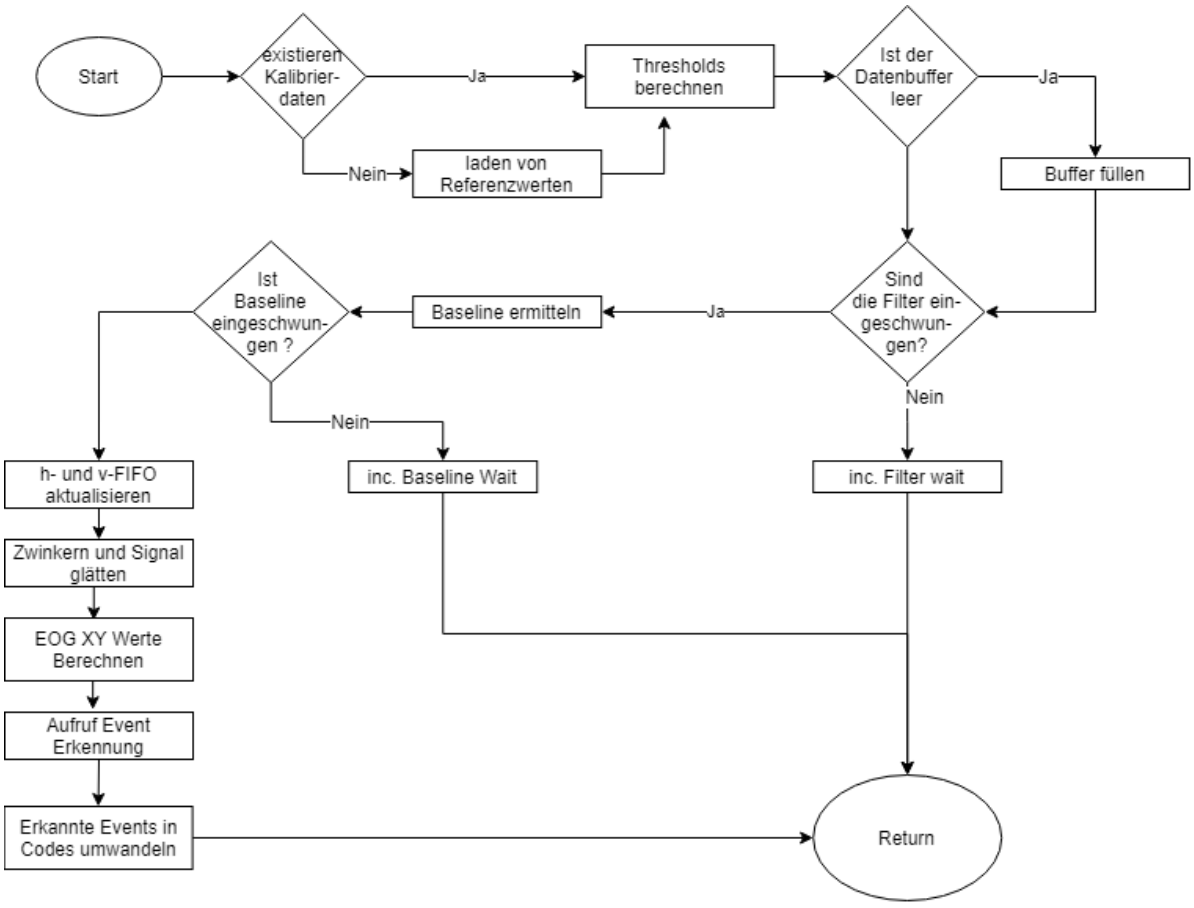


Abbildung 6-11: Funktionsablauf der Auswertefunktion

letzter Schritt vor der Erstellung der Grafik, wird eine 50Hz Filterung durchgeführt. Der Funktionsablauf der Auswertefunktion ist in Abbildung 6-11 zu sehen. Am Beginn der Auswertung werden Thresholds berechnet, danach müssen die Filter einschwingen. Anschließend wird noch die Baseline ermittelt, damit die Auswertung gestartet werden kann. Sowohl die Thresholds als auch die Baseline und die Buffer sind als persidente Variablen definiert. Als erstes wird das Signal geglättet und das Zwinkern entfernt. Mit dem „Median“ – Filter werden Störsignale, die nur 0,25 Sekunden dauern, wie beispielsweise Sakkaden oder Zwinkern entfernt. Mit Hilfe der Kalibrierungsdaten und der Baseline wird die XY Position der Blickrichtung berechnet. Danach wird die Eventerkennung gestartet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

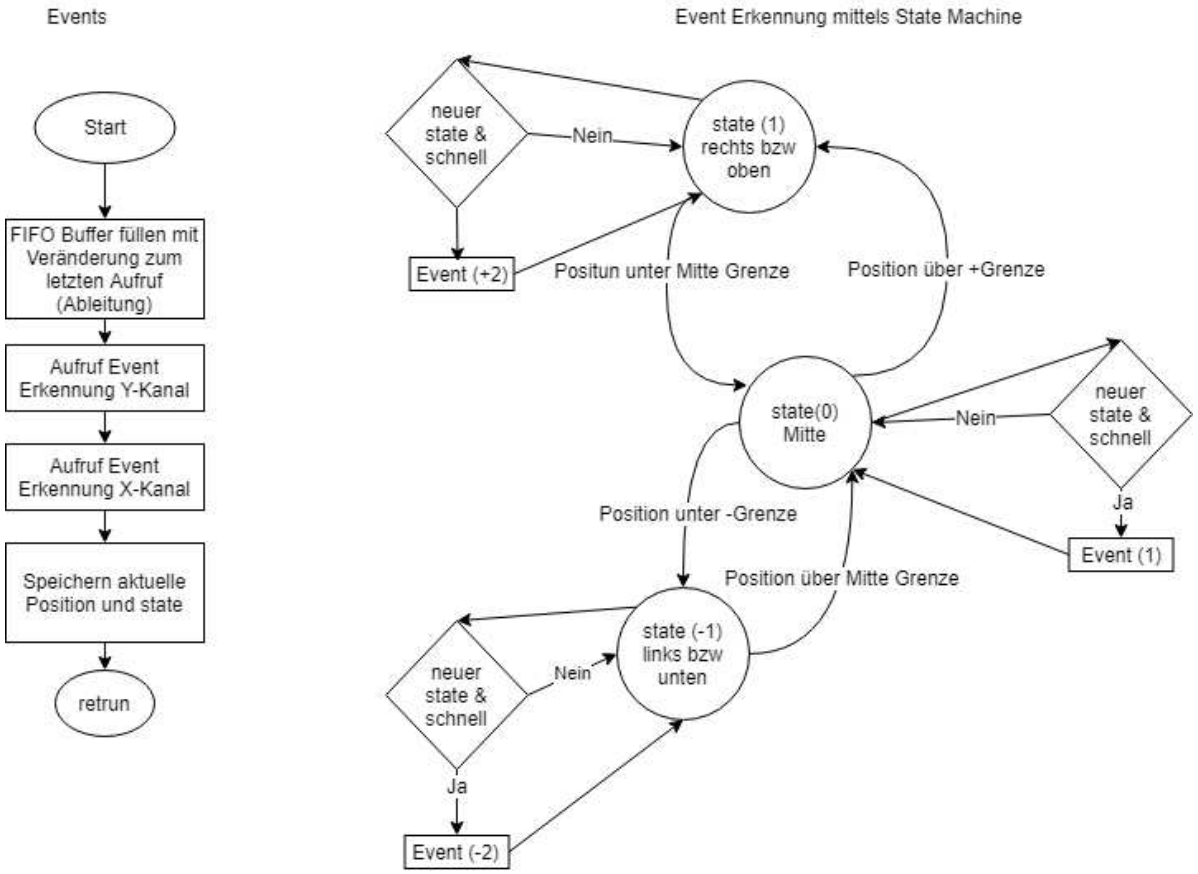


Abbildung 6-12: Funktionsablauf und Automat der Event Erkennung

Der Funktionsablauf ist in Abbildung 6-12 dargestellt. Zu Beginn werden die Veränderung zu dem letzten Werte in einen FIFO Buffer geschrieben. Die Veränderung wird über Bildung der Ableitung ermittelt. Danach wird mittels Zustandsautomaten (engl. State Machine) ermittelt, ob ein Event aufgetreten ist oder nicht. Es wird derselbe Automat verwendet, sowohl für die horizontalen als auch die vertikalen Events. Im Folgenden wird der Automat für vertikale Events beschrieben. Für den horizontalen Automaten kann oben mit rechts ersetzt werden und unten mit links. Der Automat besteht aus den drei Zuständen Oben, Mitte und Unten. Für die Erkennung, ob sich der Zustand geändert hat, gibt es jeweils eine untere und obere Grenze, wie in Abbildung 6-13 dargestellt. Damit wird verhindert, dass ein zu häufiges Springen zwischen den States vorkommt. Bei jeder Zustandsänderung wird mittels Ableitung ermittelt, ob sie schnell oder langsam erfolgt ist. Wenn es eine Zustandsänderung gegeben hat und diese schnell erfolgt ist, wird ein Event erkannt und zurückgegeben. Eine Auflistung aller möglichen Events ist in

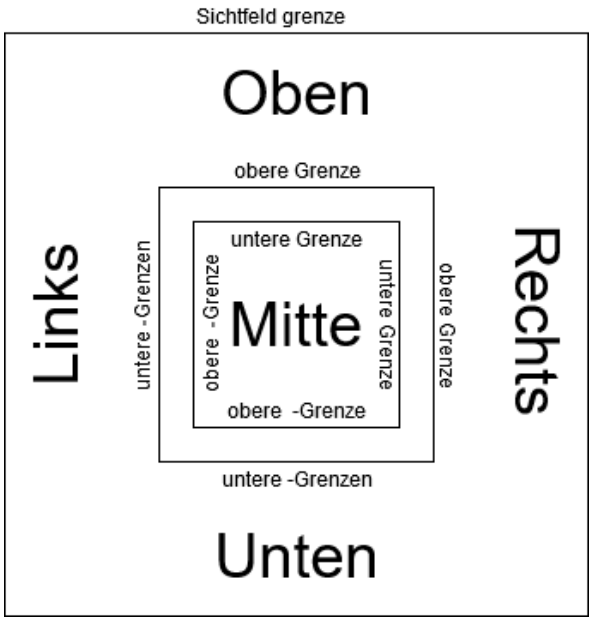
Event Nummer	Horizontales Signal	Vertikales Signal
-2	to right („tr“)	to top („tt“)
1	from left („fl“) or from right („fr“)	from top („ft“) or from bottom („fb“)
2	to left („tl“)	To bottom („tb“)

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Tabelle 6-2 aufgelistet. Events sollen nur ausgelöst werden, wenn eine eindeutige Änderung der Blickrichtung erfolgt. Die obere Grenze gilt immer, wenn eine Änderung von der Mitte ausgehen gemacht wird. Die untere Grenze gilt immer, wenn eine Blickänderung zur Mitte gemacht wird. Bei einem Event zur Mitte hin (Event 1) wird unterschieden ob vom Zustand -1 oder +1 der Zustand 0 erreicht wird.
Damit der Zustandsautomat erkennen kann ob sich ein Zustand geändert hat oder nicht wird der aktuelle diskretisierte Werte mit den Werten aus der Kalibrierung verglichen. Die Erkennung ob die Zustandsänderung langsam oder schnell ist, wird das Ergebnis der Ableitung mit dem Schwellenwert aus den Kalibrierungsdaten verglichen.

Event Nummer	Horizontales Signal	Vertikales Signal
-2	to right („tr“)	to top („tt“)
1	from left („fl“) or from right („fr“)	from top („ft“) or from bottom („fb“)
2	to left („tl“)	To bottom („tb“)

Tabelle 6-2: Liste der möglichen Events



calibdata_v_abs_top
calibdata_v_abs_bottom
calibdata_v_border_top
calibdata_v_border_bottom
calibdata_v_diff_lower_up
calibdata_v_diff_lower_down
calibdata_v_diff_upper_up
calibdata_v_diff_upper_down
calibdata_h_abs_left
calibdata_h_abs_right
calibdata_h_border_left
calibdata_h_border_right
calibdata_h_diff_lower_up
calibdata_h_diff_lower_down
calibdata_h_diff_upper_up
calibdata_h_diff_upper_down

Abbildung 6-13: Grenzen für Änderung des Zustandsautomaten und Übersicht der Kalibrierungsvariablen

6.4.1 MATLAB® Funktionen

Eine genaue Beschreibung aller verwendeten Funktionen können in der Matlab® online Hilfe gefunden werden [49]. Folgende Funktionen wurden verwendet:

- Butter: erstellt ein Butterworth Filter auf dem Datenfeld aus

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

- Filter: führt eine Filterung durch
- Medianfilter: Der Median-Filter entfernt Ausreißer in Messreihen [50].
- Repmat: Erzeugt ein Datenfeld mit identen Einträgen
- Round: Mit der Funktion Round wird der Wert auf die nächste Integerzahl gerundet.
- Zeros: erzeugt ein Feld, welches mit lauter Nullen gefüllt ist

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

7. Verifizierung und Valedierung

In diesem Kapitel werden die einzelnen Entwicklungen dieser Arbeit mit Hilfe von Test und Messungen verifiziert. Es wird der Test sowie die benötigte Test- und Messausrüstung beschrieben und die daraus resultierenden Ergebnisse. Die Validierung erfolgt nach der Verifizierung. Zuerst wird die Firmware verifiziert und getestet. Den Abschluss des Kapitels bildet die Verifizierung des Algorithmus aus dem Dipol-Augenmodell. Mit Hilfe von Daten die während der Validierung der Android-App entstanden sind werden Auswertungen des Algorithmus betrachtet.

7.1 Biosignalerfasser Software

Im Rahmen dieser Arbeit wurde die Firmware des Biosignalerfassers adaptiert. Es ist nun möglich eine bidirektionale Verbindung mit dem Biosignalerfasser mittels Terminalprogrammes herzustellen. Der Biosignalerfasser muss zuerst mit Windows gekoppelt werden, danach kann in den Eigenschaften

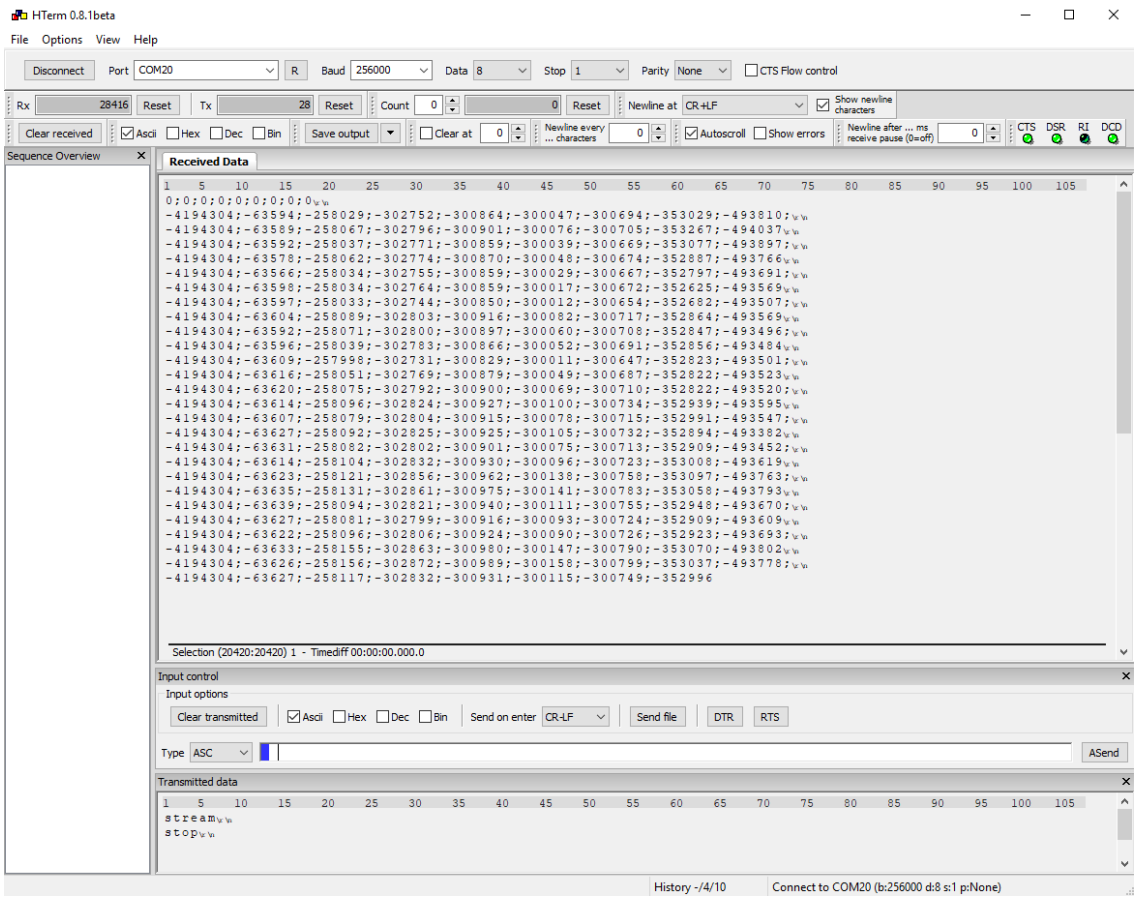


Abbildung 7-1: Screenshot starten und stoppen einer Messung

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

des Bluetooth-Gerätes der COM Port nachgeschaut werden, der dem Device zugeordnet wurde. Mit Hilfe des Terminalprogrammes kann der COM Port geöffnet werden. In Abbildung 7-1 ist zu sehen wie eine Messung mit dem Befehl „stream“ gestartet wird und dann mittels des Befehles „stop“ gestoppt wird. Die Einstellungen des COM Port sind mit 256000 Baud, 8 Data, 1 Stop und keine Parität gewählt. Rechts unten ist zu erkennen, dass eine Verbindung zur Schnittstelle COM20 aufgebaut werden konnte. Im Feld „Transmitted data“ sind die gesendeten Befehle und im Feld „Received Data“ die Empfangen Pakte. In Abbildung 7-2 sind zwei weite Testergebnisse zu sehen. In der linken Abbildung

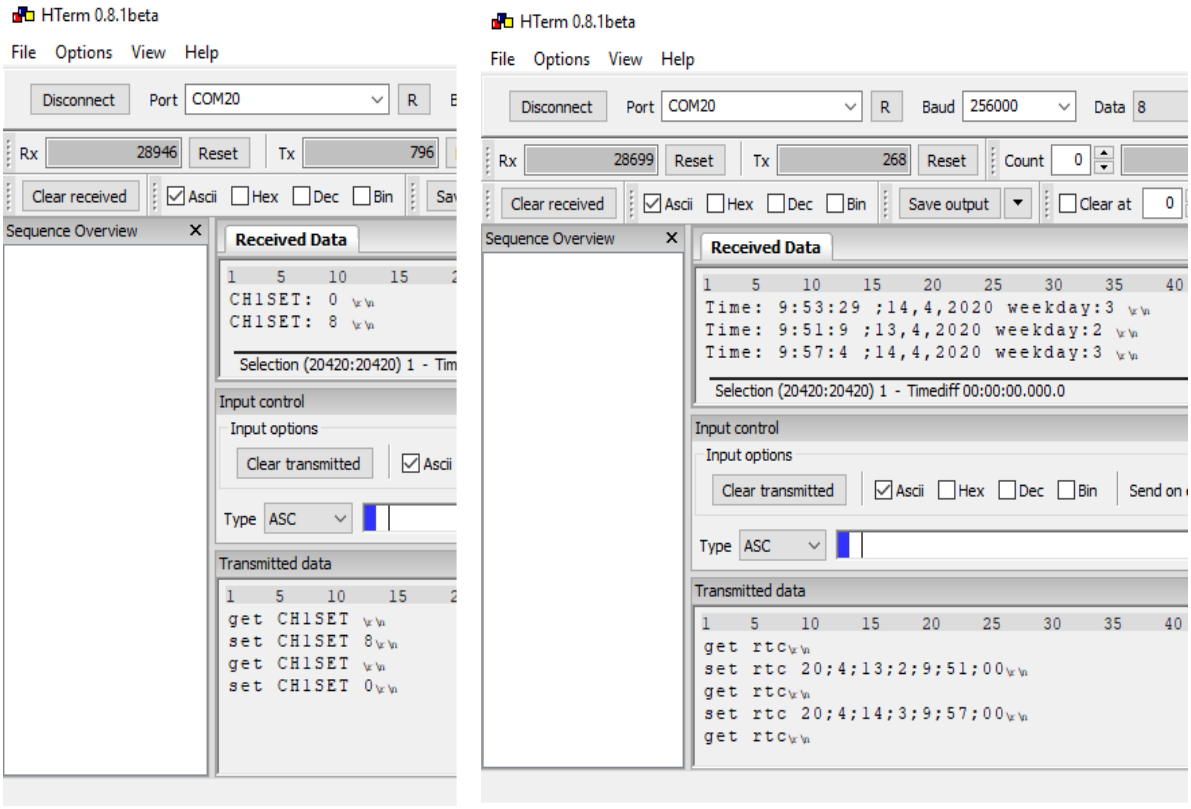


Abbildung 7-2: links setzen der Parameter für CH1; rechts setzen der RTC

wurde das Steuerregister vom ersten Kanal des Biosignalerfassers geändert. Es wurde die Verstärkung vom Standartwert 6 auf 1 geändert. Zuerst wurde der aktuelle Wert mittels des Befehles „get CH1SET“ abgefragt. Der Biosignalerfasser hat die erwartete Antwort mit 2CH1SET: 0 \r\n2 geliefert. Um dem Verstärkungsparameter zu ändern muss das Register „CH1SET“ auf den Wert 8 (binär 00001000) gesetzt werden, dazu ist der Befehl „set CH1SET 8\r\n“ an den Biosignalerfasser gesendet worden. Anschließend wurde der Wert für das Register abgefragt und die Antwort mit der Erwartung überprüft. In der rechten Abbildung von Abbildung 7-2 ist zu erkennen, wie die RTC neu gesetzt worden ist. Auch hier wurde zuerst der aktuelle Wert der RTC mittels des Befehls „get rtc\r\n“ abgefragt. Danach wurde die rtc auf den vorigen Tag mit dem Befehl „set rtc 20;4;13;2;9;51;00\r\n“ gesetzt. Durch eine Abfrage der aktuellen Zeit auf der RTC wurde die Antwort mit der erwarteten Antwort wieder überprüft. Auch mit einer SPP Bluetooth-App auf einem Mobilen Device kann eine Ver-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

bindung mit dem Biosignalerfasser aufgebaut werden. Auf dem Tablet wurde mit Hilfe des Programmes TerminalBT in der Version 1.0 eine Verbindung mit dem Biosignalerfasser hergestellt. Nach „Sent:“ steht immer der Befehl der gesendet wurde und nach dem „>“ steht immer die empfange Nachricht. Zu erkennen ist, dass zuerst wieder die Settings des ersten Kanals abgefragt wurden, danach

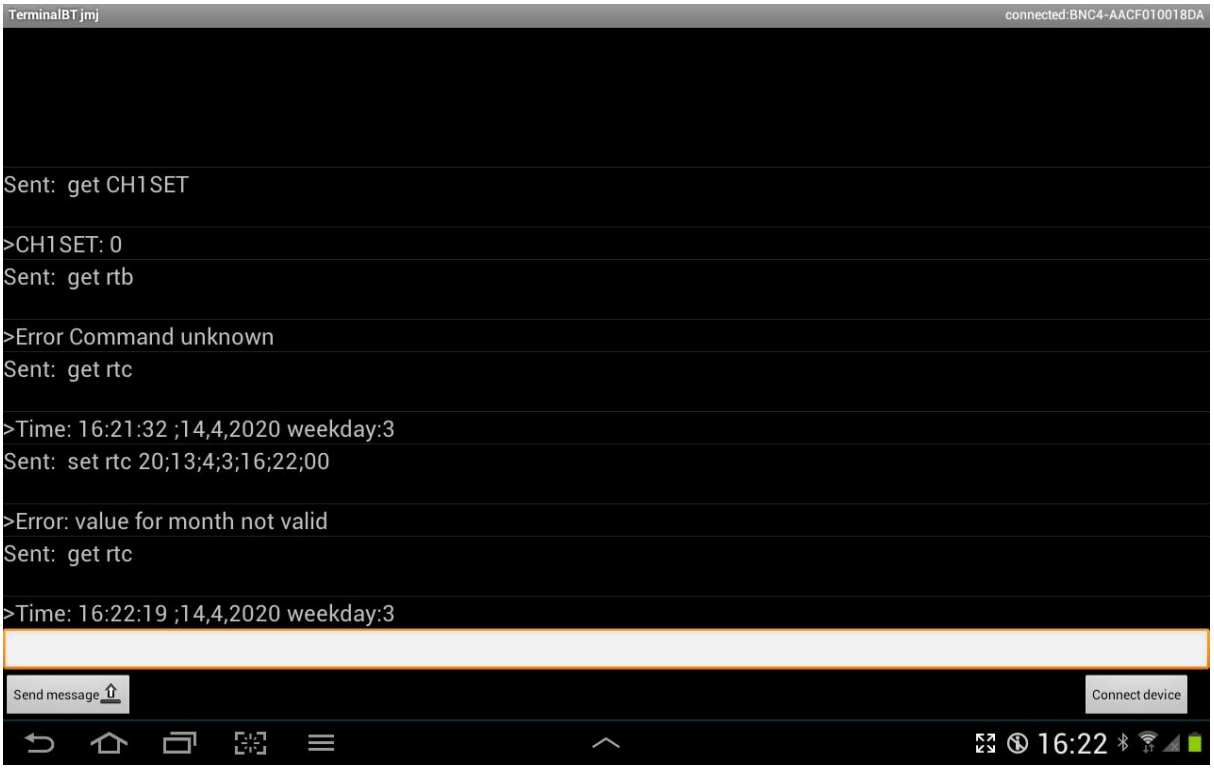


Abbildung 7-3: Screenshot der Android App TerminalBT; Fehlermeldungen bei fehlerhaften Befehlen

wurde ein fehlerhafter Befehl gesendet. Der Befehl „get rtb“ ist im Übertragungsprotokoll nicht definiert, daher gibt der Biosignalerfasser die Fehlermeldung „Error Command unknown“ zurück. Anschließend wurde der aktuelle Wert der RTC abgefragt. Um zu testen ob die RTC mit einem falschen (weil nicht möglich) Wert gesetzt werden kann wurde als Test ein „set rtc“ Befehl gesendet. Für den Monat der neuen Zeit wurde der Wert 13 angegeben. Der Biosignalerfasser erkannte den Fehler und meldete den ersten Fehler den er detektiert hat mit der Nachricht: „Error: value for month not valid“. Um zu überprüfen ob die RTC noch korrekt gesetzt ist, wurde nochmal der aktuelle Wert abgefragt. Wie vorgesehen hat der Biosignalerfasser keine Änderung an der Zeiteinstellung der RTC vorgenommen.
Als Test für die „record“ Funktion wurde über die Terminal Verbindung der Befehl „record“ an den Biosignalerfasser gesendet. Nach Beendigung der Bluetooth Verbindung startet die Aufzeichnung auf die SD Karte, die im Biosignalerfasser eingelegt wurde.

Mithilfe dieser und weiteren Test wurde die Validierung der Firmware durchgeführt. Die Verifizierung der einzelnen neuen Softwarekomponenten wurde mithilfe der Vorgaben von AIT unter Mithilfe der Enwickungsumgebung durchgeführt. Ein Release Protokoll der neuen Firmware ist im Anhang zu finden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

7.2 Android® Applikation

Um die entwickelte Android Software verifizieren zu können, werden die verschiedenen Aufgaben der Applikation nach einander getestet. Im Folgenden sind Screenshots der Applikation „KlaVigView“ gezeigt, die Einstellungen und Messungen zeigen. Alle Screenshots und Test mit Android wurden auf einem Samsung Tablet vom Modell GT-P7501 durchgeführt. Die Validierung der entwickelten Software wurde ebenfalls mit dem Tablet durchgeführt.

7.2.1 Verbindungsaufbau

Wenn die Applikation gestartet wird kann über den Button “Disconnect“ oder mittels des Settings Menü im Punkt „Bluetoot settings“ das Bluetooth-Device ausgewählt werden. In Abbildung 7-4 ist ein Screenshot der Applikation zu sehen. Im Vordergrund ist der Android Dialog zu erkennen, über den das Bluetooth Geräte sucht bzw. ein bereits bekanntes Gerät ausgewählt werden kann. Im Hintergrund ist das Settings Menü zu erkennen, welches im Abschnitt 6.3.4 beschrieben wurde.

Abbildung 7-4: Screenshot der KlaVigView App; Verbindungsaufbau zum Bluetooth-Device

Sobald ein Gerät ausgewählt wurde, versucht die Anwendung sich mit diesem Gerät zu verbinden. Wenn die Verbindung erfolgreich war, wird sofort die Messung gestartet. Es wird standartmäßig der gemessene Signalverlauf des ersten Kanales angezeigt. Sollte keine Verbindung möglich sein wird eine Meldung eingeblendet und es muss der Vorgang zum Aufbau der Bluetooth Verbindung, mit der Auswahl eines Gerätes, neu gestartet werden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

7.2.2 Verifikation der Software

Um die entwickelte Software verifizieren zu können ist ein definiertes Signal in den Biosignalerfasser eingespielt worden. Der Messaufbau ist in Abbildung 7-5 dargestellt. Als Signalquelle und als Oszil-

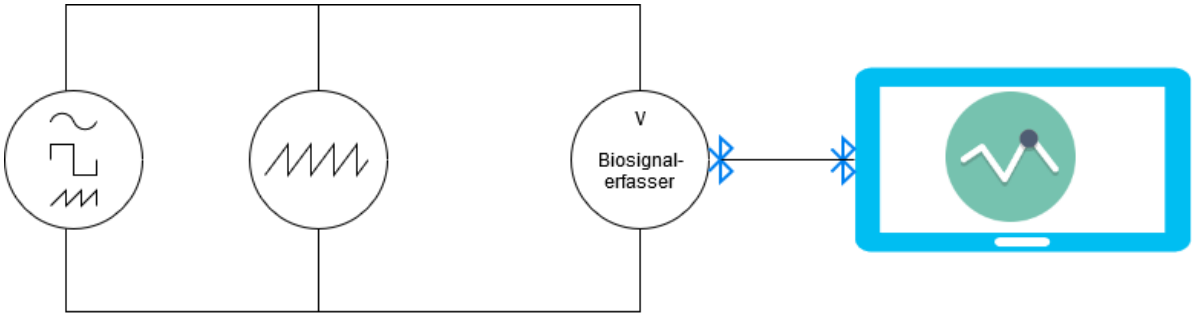


Abbildung 7-5: Messaufbau Verifikation Android App

loskope ist ein MSOX3034T von Keysight zum Einsatz gekommen. Das Signal wurde mittels abgefederten Klemmhaken am Biosignalerfasser eingespielt. Als Signal wurde ein Sinus mit einer Amplitude von 35 mVolt und 1Hz eingespeist. Die Einstellungen am Oszilloskop wurden so gewählt, dass das Signal gut ablesbar dargestellt wird. Ein Schreenshot vom Oszilloskop ist in Abbildung 7-6 zu sehen. Das Signal wurde vom Biosignalerfasser gemessen und die Messwerte mittels Bluetooth an ein Samsung GT-P7501 Tablet übertragen. Der Verstärkungsfaktor wurde auf eins gestellt. Am Tablet wurden die Daten von der im Rahmen dieser Arbeit entwickelt App „KlaVigView“ empfangen und

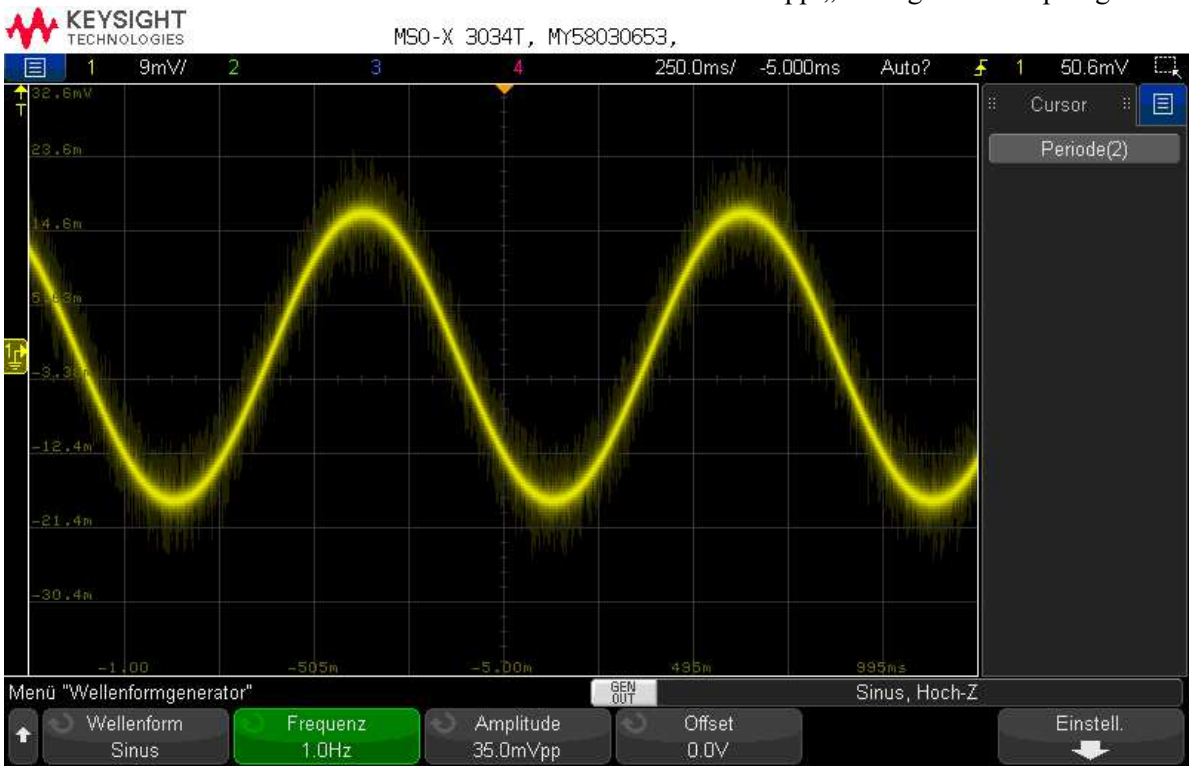


Abbildung 7-6: Messung des Testsignals mittels Oszilloskope

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

dargestellt. Bei der Android-App soll vor allem der Signalverlauf optisch kontrolliert werden. Wie Abbildung 7-7 ist der Sinus sehr gut zu erkennen. Für eine Verifizierung ist wichtig, dass die gemessenen Werte auch den tatsächlichen Werten entsprechen. Wie in Kapitel 6.3.2 beschrieben, werden 250 Messpunkte in der Android App dargestellt. Damit nicht nur die letzte Sekunde des gemessenen Signals am Display zu sehen ist, wird nur jeder dritte Messwert genommen. Daher sind drei Sekunden am Display zu sehen. Wie zu erkennen ist, sind 3 Perioden des 1 Hz Sinus zu erkennen.

Um die Amplitude berechnen zu können muss der Maximale und der Minimale ADC Wert addiert

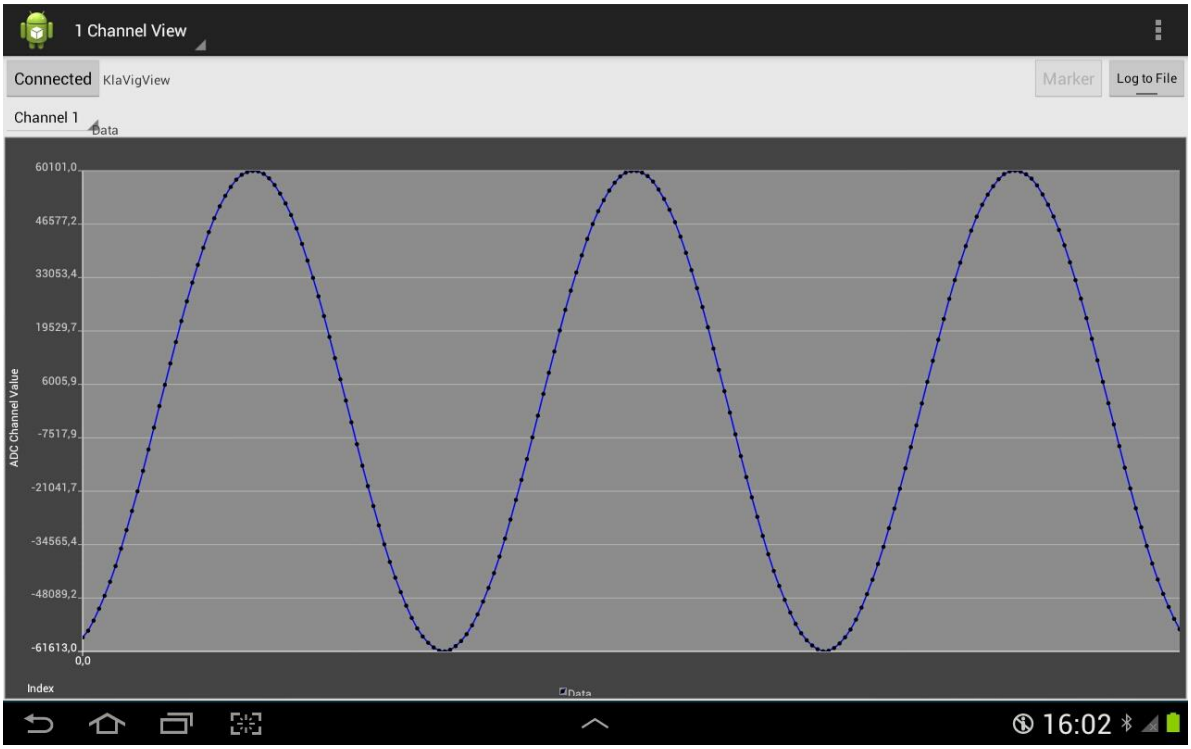


Abbildung 7-7: Screenshot aus der Android App

werden. Der Biosignalerfasser hat eine Betriebsspannung von 3,3V. Damit ergibt sich laut Datenblatt [42] eine Referenzspannung von 2,5 Volt. Der Biosignalerfasser hat eine Auflösung von 24 Bit. Mit Hilfe der Formel 7.1 lässt sich die gemessene Spannung berechnen. Der gemessene Wert beträgt 36,2 mV. Wie am Oszilloskope zu erkennen ist, ist das Signal von Störquellen überlagert. Daher misst der Biosignalerfasser korrekt.

$$V = \left(\left|ADC_{posetiv}\right| + \left|ADC_{negativ}\right|\right) * \frac{V_{Ref}}{(2^{Resulution} - 1)\, ADC_{Gain}} \qquad (7.1)$$

Derselbe Versuch wurde mit dem Verstärkungsfaktor sechs am Biosignalerfasser durchgeführt und lieferte dasselbe Ergebnis. Die Verzögerung, die sich über die Datenübertragung bei Bluetooth ergibt, ist variabel und daher schwer zu messen. Der interessante EOG Frequenzbereich umfasst nur Frequenzen bis 10 Hz. Daher wurde auf die Erstellung eines Bode-Diagrammes verzichtet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

7.2.3 Validierung der Software

Für die Validierung der adaptierten Firmware und der Android Applikation wurden Messungen unter realen Bedingungen durchgeführt. So konnten sich die Probanden während der gesamten Messung frei bewegen. Es sind während der Messung verschiedene Augenbewegungen vorgegeben worden und das Signal wurde aufgezeichnet und damit es später als Input für die Klassifizierung verwendet werden konnte. Damit die Messdaten besser zugeordnet werden konnten, wurde auch ein „Marker Button“ implementiert. Sobald eine Aufzeichnung der EOG Daten beginnt, werden die Daten in einem File auf dem Android Device gespeichert. Sobald auf den „Marker“ Button gedrückt wird, wird am Ende des letzten empfangen Datenframes das Wort Marker mit einer laufenden Nummer hinzugefügt. Diese Notation ist im Datensatz sehr leicht zu finden. Somit kann der Anfang bzw. das Ende einer Aufgabe

Abbildung 7-8: vertikales EOG Signal

bei der Auswertung im Datenfile gefunden werden und nur dieser Teil ausgewertet werden.
Wie in Kapitel 6.3.3 beschrieben, gibt es drei verschiedene Graphen in der Androird App in Abbildung 7-8 ist eine Messung eines horizontalen EOG Signals dargestellt. Die Aufgabe des Probanden war es, abwechselnd nach links und nach rechts zu schauen. Das dargestellte Signal zeigt den nach dem Dipol Modell zu erwarteten Verlauf. In Abbildung 7-9 ist der Verlauf von zwei Kanälen gleichzeitig dargestellt. Im oberen Graphen ist der Verlauf des horizontalen Signals und im unteren Graphen der vertikale Verlauf. Die Aufgabe des Probanden war es, den größtmöglichen Kreis mit den Augen zu beschreiben. Die letzte Ansicht ist in Abbildung 7-10 zu sehen und zeigt das Ergebnis, wenn die Messwerte dem Signal entsprechend auf den Achsen aufgetragen werden. Auch bei dieser Messung war die Aufgabe des Probanden einen größtmöglichen Kreis mit den Augen zu beschreiben. Das allgemein

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

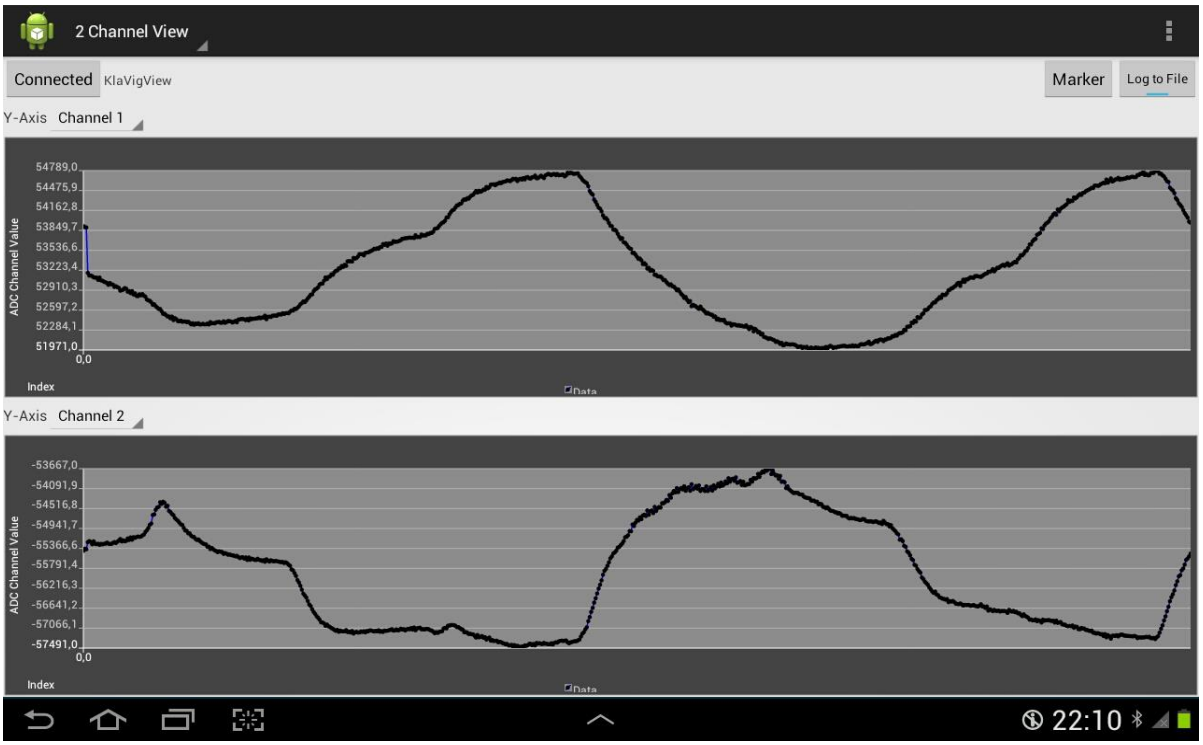


Abbildung 7-9: horizontales und vertikales EOG Signal

erwartete Ergebnis ist vermeintlich ein Kreis. Es ist jedoch eine Ellipse zu erkennen, dies kommt daher, dass für das horizontale Signal, die Potentailverschiebung beider Augen, gemessen wird, hin-

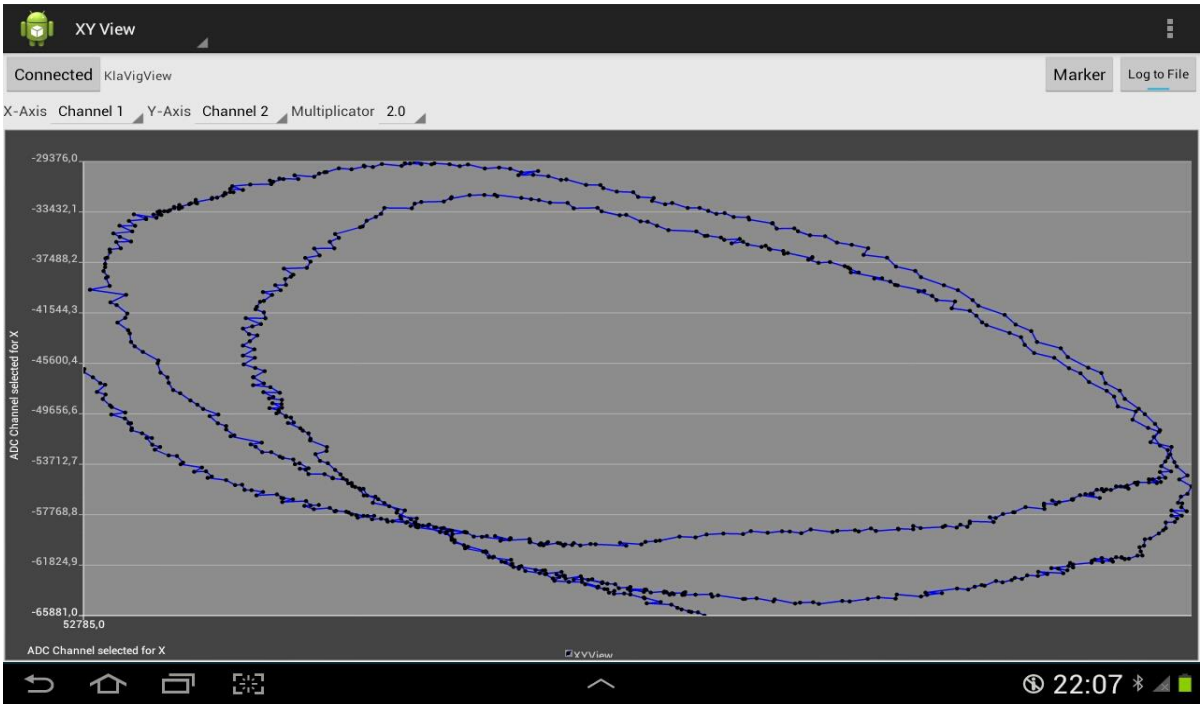


Abbildung 7-10: XY View, horizontaler Kanal auf der X-Achse, vertikaler Kanal auf der y Achse

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

gegen für das vertikale Signal nur von einem Auge. Mit Hilfe der Multiplikator Option kann das Verhältnis besser angepasst werden. Dennoch ist die Kurve eine Ellipse. Das erklärt sich dadurch, dass das Sichtfeld des Menschen in horizontaler Ausdehnung 180° hingegen in vertikaler Richtung jedoch nur 60° nach oben und 70° nach unten beträgt. Aus diesem Grund entspricht der gemessene Signalverlauf dem erwarteten Signal.

7.3 Ergebnisse der Matlab-Signalverarbeitung

Die mittels Android App aufgezeichneten Messdaten von EOG Signalen werden als Eingabedaten für die Matlab Signalverarbeitung verwendet. Das aufgezeichnete Signal beinhaltet genau vorgegebene Augenbewegungen, um später das Ergebnis interpretieren zu können. Die Daten werden mit Hilfe des Matlab Befehls „dmlread“ aus ASCII-getrennte Daten mit numerischen Werten in eine Datenstruktur eingelesen. Danach kommt die entwickelten Matlab Scripte zum Einsatz. Die Auswertung wir mittels des Skripts „liveeog_offline_forKlavig(‘[FILENAME]‘,[ABTASTRATE])“ gestartet. Es wird gleich der Auswertebildschirm angezeigt und die erkannten Events an passender Stelle zum Signal geschrieben. Bei der offline Analyse wird am Schluss eine Übersicht der gesamten analysierten Signalssequez angezeigt. Es wird das Originalssignal, das geglättete Signal, die Zustandserkennung und Ableitung der Zustandserkennung in jedem Plot angezeigt. Die erkannten Events werden ebenfalls mit einem Kürzel am erkannten Zeitpunkt angezeigt.

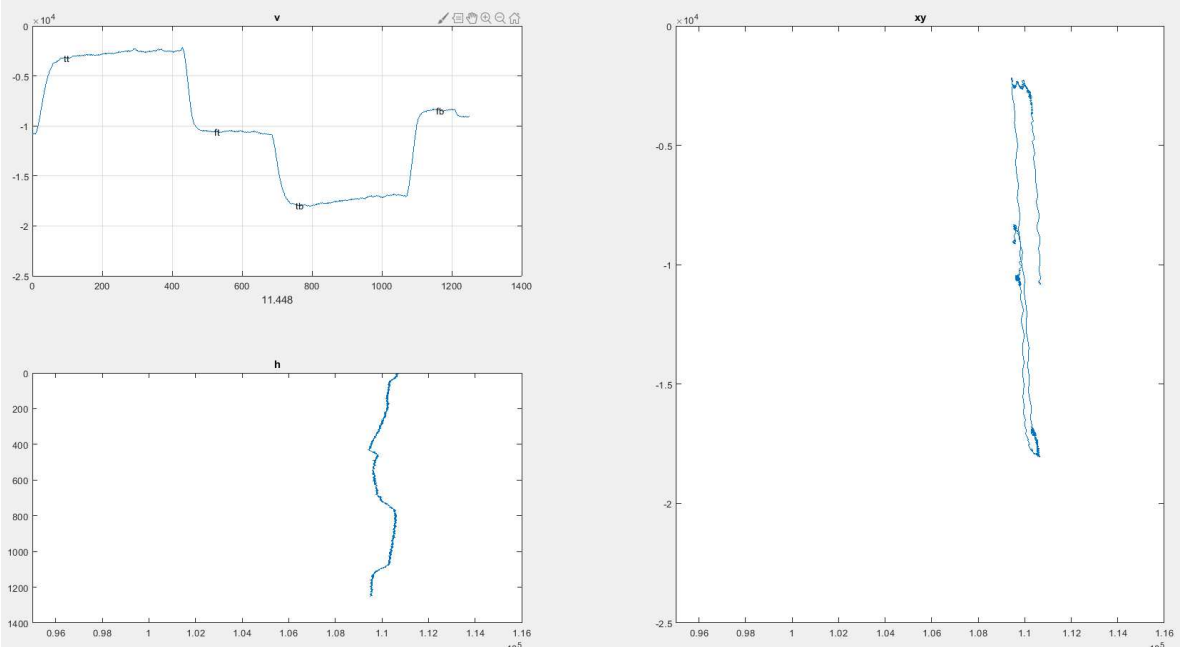


Abbildung 7-11: Matlab Ausgabe während der Analyse

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

7.3.1 EOG Event Auswertung für horizontale und vertikale EOG-Signale

Um die Auswertung horizontaler Signale zu testen, wurde ein EOG Signal aufgezeichnet der Proband sollte dabei zuerst ca. 3 Sekunden lang die Augen langsam kreisen. Damit ist sichergestellt, dass alle Einschwingvorgänge der Signalverarbeitung abgeschlossen sind. Danach sollte er nach oben schauen und die Position kurz halten um dann wieder in die Mitte zu schauen, anschließend soll das selbe mit der anderen Richtung wiederholt werden. Der Vorgang soll 5-mal wiederholt werden. Danach wird die Messung abgeschlossen. In Abbildung 7-11 ist ein Screenshot der Matlab Ausgabe während der Auswertung des Signals. Es wird immer die Auswertung der letzten 1400 Messwerte angezeigt. Der Screenshot zeigt das Signal um die Sekunde 10 der Messung an. Da der Wertebereich für das horizontale Signal und das vertikale Signal gleich groß ist, kann der Verlauf der Kurven miteinander verglichen werden. Der Plot für das vertikale Signal ist links oben zu sehen, der für das horizontale Signal links unten. Es ist zu erkennen, dass der Proband keine großen horizontalen Augenbewegungen gemacht hat, hingegen im Plot für den vertikalen Kanal sind die Augenbewegungen deutlich zu erkennen. Die Events „to top“ (tt) ,“from top“ (ft),“to bottom“ (tb), “from bottom” (fb) werden direkt im Signalverlauf eingeblendet nach Erkennung. Im rechten Plot ist ein XY Plot von beiden Kanälen gezeigt und zeigt die Augenposition im Sichtfeld des Probanden an, auch in dieser Auswertung ist zu erkennen, dass der Proband kaum horizontale Augenbewegungen gemacht hat. Die vertikalen Bewegungen der Augen sind gut zu erkennen. In Abbildung 7-12 ist die komplette Auswertung des aufgezeichneten Signals zu erkennen. Die Aufzeichnung ist 21 Sekunden lang. Der Signalausschnitt aus Abbildung 7-11 ist zwischen den Markierungen von „tt“, erstes erkanntes Event, und „fb“, letztes erkanntes Event, im Bereich zwischen Sekunde 7 und 11 in der Abbildung 7-12 zu erkennen. Event: erst wenn die Augenbewegung schnell genug ist, wird ein Event erkannt. Daher muss der Wert der Ableitung in diesem Punkt größer sein als der Vergleichswert in der Kalibrierung (Abbildung 6-13 zeigt die Variablen).

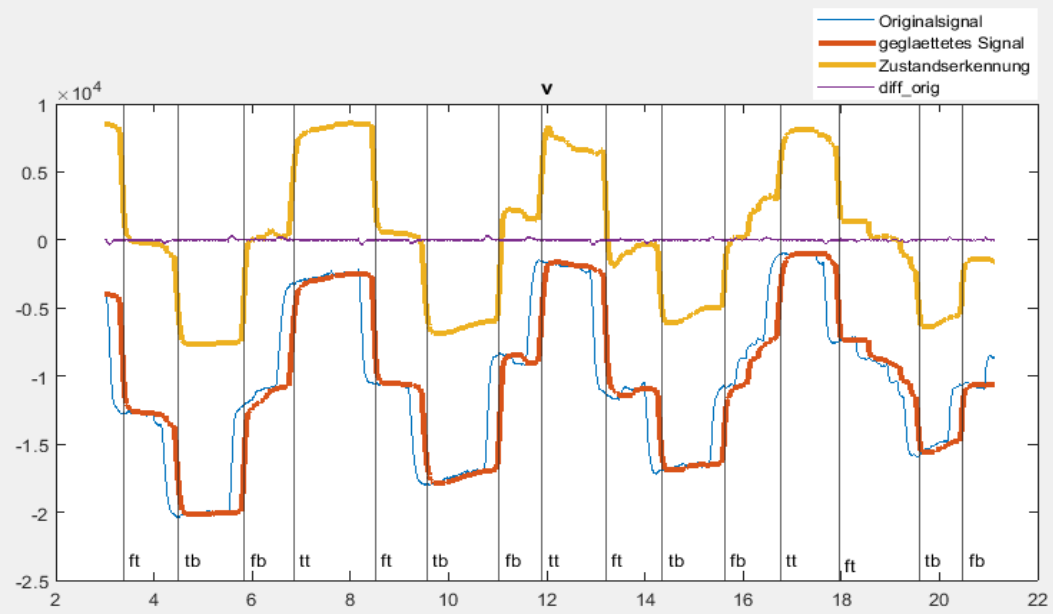


Abbildung 7-12: abgeschlossene Auswertung für ein vertikales EOG-Signals

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

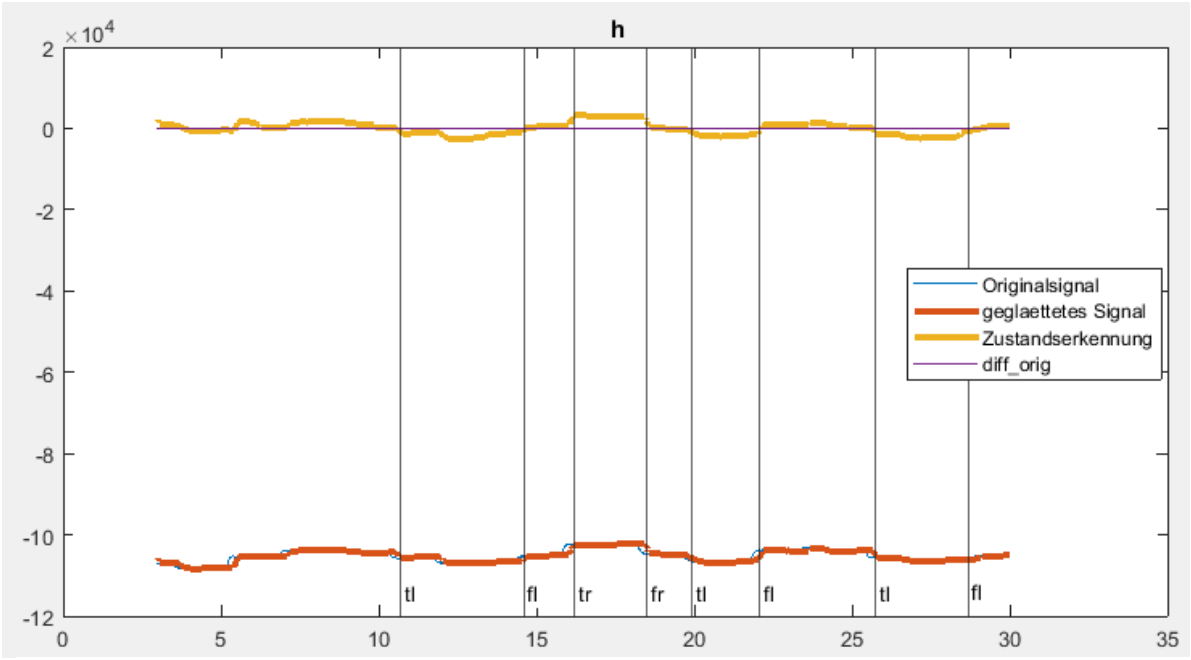


Abbildung 7-13: abgeschlossene Auswertung für ein horizontales EOG-Signal

Derselbe Test wurde für das horizontalen EOG Signal durchgeführt. Am Anfang sollte der Proband wieder einen langsamen Kreis mit den Augen beschreiben. Anschließend zu den Positionen oben, Mitte und unten schauen und jeweils kurz die Position halten. In Abbildung 7-13 ist die Auswertung für diesen Fall abgebildet. Am Anfang ist zu erkennen, dass noch keine Events erkannt werden. Erst bei der elften Sekunde wird das Event „schnell nach links schauen“ erkannt. Im Anschluss werden die nächsten Übergänge und die damit verbunden Events richtig erkannt. Bei der letzten Blicksequenz

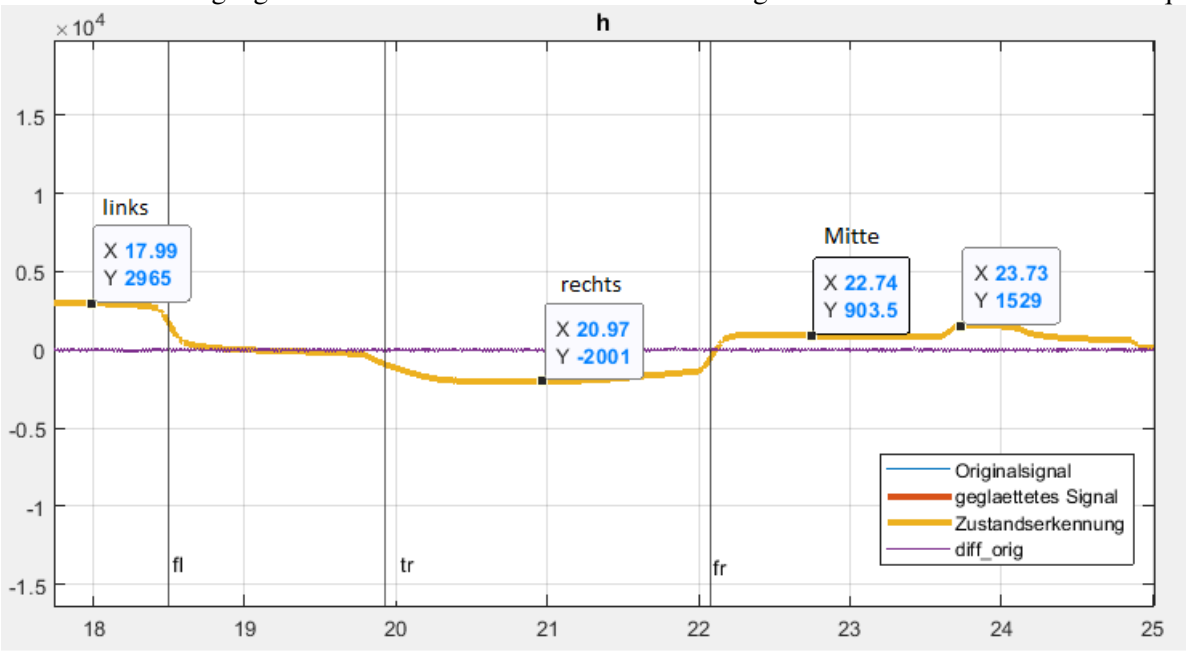


Abbildung 7-14: Detailansicht der horizontalen EOG Auswertung

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

nach oben und wieder in die Mitte wird kein Event erkannt. In den Signaldaten ist zu erkennen, dass der Proband nicht weit genug nach oben geschaut hat um hier eine Zustandsänderung im Zustandsautomaten herbei zuführen und damit ein Event auszulösen. In Abbildung 7-14 ist ein vergrößerter Ausschnitt aus der Auswertung für das vertikale EOG Signal zu sehen. Es wird nur der Signalausschnitt mit der Zustandserkennung und deren Ableitung angezeigt. Zu erkennen ist, dass die Amplitude der horizontalen Augenbewegung nicht so groß war wie bei der vergangenen Bewegung nach links. Daher hat der Algorithmus kein Event erkannt. Eine genaue Analyse der Daten ergab, dass die Position unterhalb der Schwelle für die Erkennung einer Zustandsänderung zum Zustand „Left“ befand. Die Schwelle um eine Positionsänderung nach oben zu erkennen hat in dieser Situation den Wert 1678, das Signal den Wert 1529. Daher ändert sich der Zustand nicht von Mitte nach oben.
Entscheidend für die Auswertung der EOG Signale ist die korrekte Kalibrierung. Die Kalibrierung wird am Anfang dem Skript „liveeog_getreferencecalibration.m“ aus einem File geladen. Sie unterscheidet sich jedoch für jede Aufzeichnung. Mit dem Skript „liveeog_generateReference.m“ kann eine Kalibrierung, aufgrund von Statistiken, aus einer Aufzeichnung generiert werden. Mit ihr werden die Grenzen für die Zustandsänderung und auch für die Event Erkennung festgelegt. Die Matlab Skrips können im Anhang gefunden werden.

7.3.2 EOG Auswertung für Signale aus der App Validierung

Mit der Matab-Auswertung wurden auch einzelne aufgezeichnete Signal aus der Android Software Validierung ausgewertet. So wurde auch das Signal ausgewertet, woraus der Screenshot aus Abbildung 7-10 während der Aufzeichnung gemacht wurde. Die ersten 36 Sekunden der Aufzeichnung sind

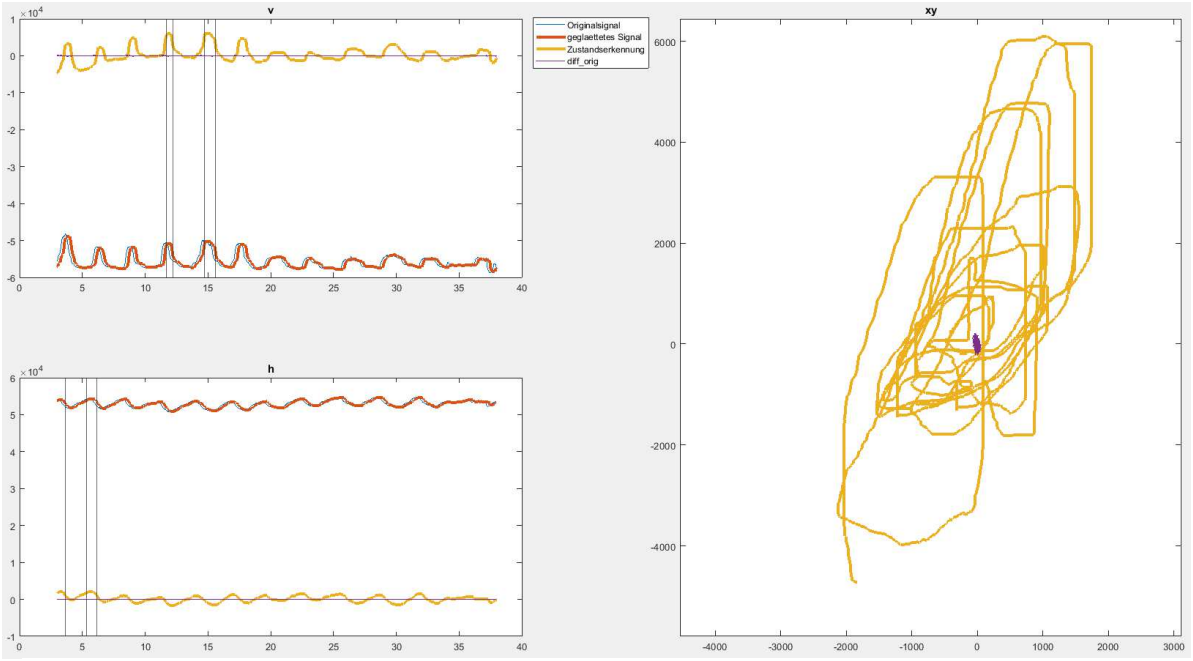


Abbildung 7-15: Matlab-Auswertung für Signalausschnitt „Augenbewegung Kreis“ aus einem Versuch

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

in Abbildung 7-15 ausgewertet worden. Zu erkennen ist, dass wenige Events, also schnelle Übergänge von einer Augenposition zu einer anderen, erfolgt sind. Da die Aufgabe des Probanden war, möglichst große Kreise zu schauen und keine schnellen Augenbewegungen zu machen, zeigt die Auswertung das erwartete Verhalten. Interessant ist, dass die Kreisbewegungen in den ersten 20 Sekunden scheinbar größer waren als danach. Da die Aufgabe war, so große Kreise wie möglich mit den Augen zu beschreiben, ist die Frage, ob der Proband die Augenbewegungen nicht mehr so exakt ausgeführt hat, oder ob sich etwas in der Messkette verändert hat und daher geringere Amplituden gemessen wurden. Die Frage, die sich nur für den Auswerter und damit auch für den Algorithmus ergibt, ist muss die Kalibrierung angepasst werden oder nicht. Es gibt eine Vielzahl von möglichen Ursachen die, innerhalb oder außerhalb des Einflussbereiches des Systems liegen. Eine Veränderung des Übergangswiederstandes der Elektroden, eine Änderung der Lichtverhältnisse oder Ermüdung der Augenmuskulatur des Probanden könnten alle verantwortlich sein für die Änderung im Signal. Die Kalibrierungsdaten müssten, abhängig von der Ursache geändert werden oder nicht. Noch deutlicher sichtbar wird die Auswirkung von externen Einflüssen, wenn das Signal länger betrachtet wird. In Abbildung 7-16 ist die Dauer der Auswertung für das Signal für 75 Sekunden dargestellt. Die großen Sprünge im Signal lassen sich nur mehr durch Veränderungen des Übergangwiederstandes an den Elektroden erklären. Diese können ausgelöst werden durch Kopfbewegungen, die durch ein Ziehen oder Drücken der Kabel auf die Elektroden eine Veränderung herbeiführen oder durch Bewegungen der Gesichtsmuskulatur, wie z.B. Lachen den Übergangswiederstand der Klebelektroden verändert. In der Abbildung 7-16 ist weiter zu erkennen, dass das Signal nach den Sprüngen, durch die Filterung wieder in einem normalen Wertebereich zurückkommt und die Erkennung der Zustände und der Events danach wieder funktioniert. Die Events und Zustandsänderung bei zu großen Sprüngen im Signal müssen im Algorithmus noch gesondert behandelt werden.

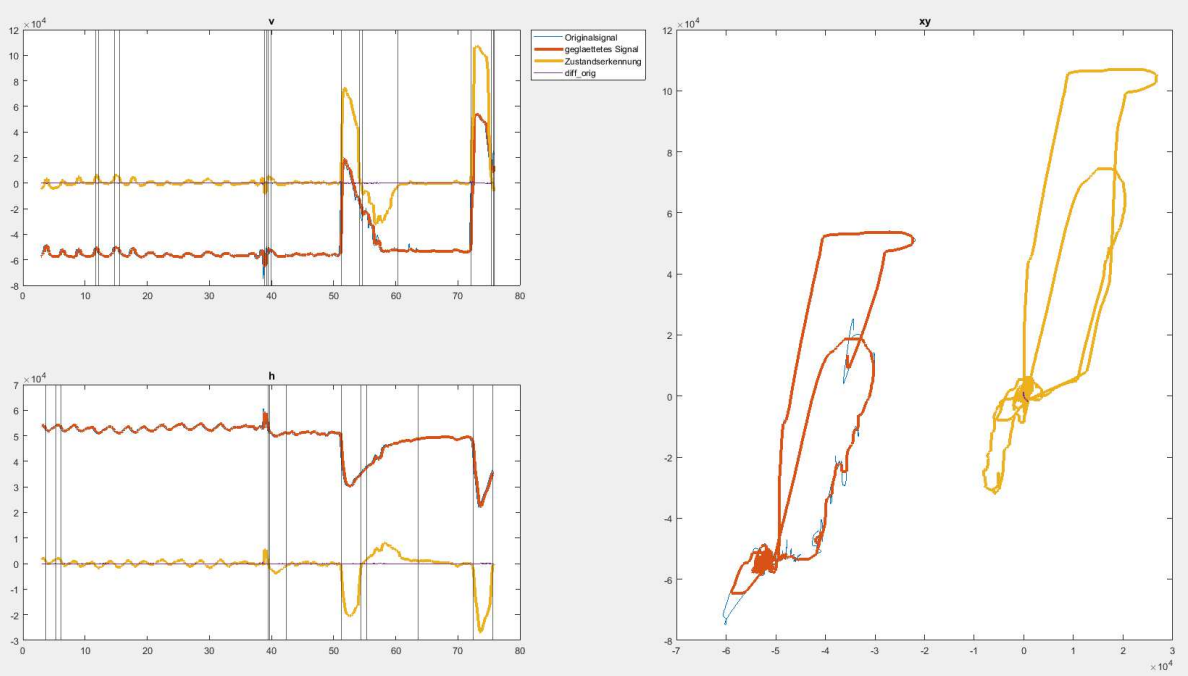


Abbildung 7-16: EOG Signal mit Messstörungen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

7.3.3 EOG-Signals live Auswertung

Der Algorithmus und die Kalibrierung wurden mit Hilfe der aufgezeichneten Signale des Biosignalerfassers entwickelt und getestet. Um auch die Möglichkeit aufzuzeigen, dass der Algorithmus schnell genug arbeitet um die EOG-Signale in Echtzeit auszuwerten, wurden auch Test mit einer Live Auswertung des EOG Signals durchgeführt. In Abbildung 7-17 ist der Messaufbau, mit dem ein Referenz Signal über den Biosignalerfasser in den Auswertealgorithmus eingespeist wurde, dargestellt. Das Referenz Signal war ein Sinus mit einer Amplitude von 35 mV und einer Frequenz von 1 Hz. Es handelt

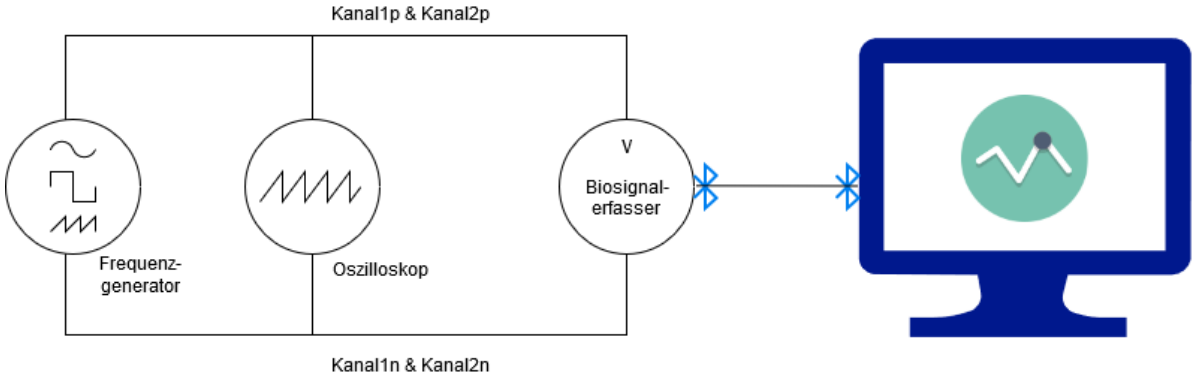


Abbildung 7-17: Messaufbau für Verifikation der Matlab-Live Messung

sich um denselben Messaufbau wie beim Punkt 7.2.2, nur dass die Daten vom Biosignalerfasser nicht mehr an die Android App übermittelt werden, sondern an den Matlab Computer. Über die Formel 7.1 lässt sich der gemessene Spannungswert bestimmen. Die Frequenz kann über die Anzahl der gezeigten Messpunkte und die Abtastrate bestimmt werden. Bei einer Anzeige der letzen 5000 Punkte mit einer Abtastrate von 250Hz werden die letzten 20 Sekunden am Bildschirm angezeigt. In Abbildung 7-18 sind sowohl beim horizontalen wie auch beim vertikalen Signal 20 Schwingungen zu erkennen. Da das Eingangssignal sowohl für Kanal1 als auch für Kanal2 das gleiche ist, sind die beiden Signale phasengleich. Daher ergibt sich für die XY Auswertung eine Gerade von rechts oben nach links unten. Anders als in der Android Applikation sind hier keine Störungen zu sehen. Durch die Filterung im Algorithmus werden diese entfernt.
Wie schon in der „offline“ Auswertung gezeigt wurde, funktioniert die Erkennung der Events sowohl für das vertikale als auch das horizontale Signal. Der dahinterstehende Algorithmus ist auch derselbe. Um die Übersichtlichkeit zu wahren wird bei der „live“ Auswertung nur die Auswertung für das horizontale Signal betrachtet. Wie aus den anderen Messungen erkenntlich ist der Wertebereich des horizontalen Signals kleiner.

Ein Screenshot einer „online“ Auswertung für ein horizontales EOG Signal ist in Abbildung 7-19Abbildung 7-18 zu sehen. Die Aufgabe des Probanden war es, wieder abwechselnd nach links, in die Mitte und nach rechts zu schauen und jeweils kurz die Position zu halten. Die Übergänge von den verschiedenen Augenpositionen sollte dabei so rasch als möglich erfolgen. Zu erkennen ist, dass bei allen Änderungen der Blickrichtung erwartungsgemäß ein Event erkannt wurde. Das Matlab Skript

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

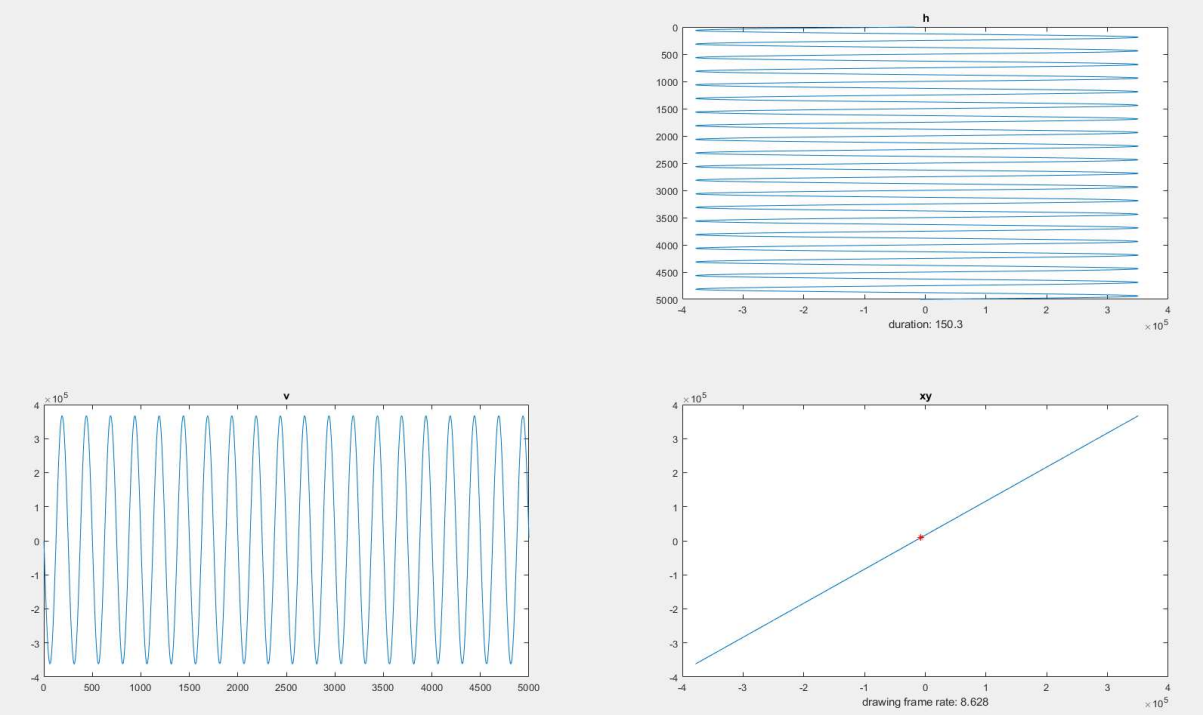


Abbildung 7-18: Übersicht der live EOG Auswertung in Matlab

schreibt die Events direkt an die Stelle, an der sie erkannt wurden, und damit direkt in den Signalverlauf. Damit die Lesbarkeit erhöht wird, wurden die Beschriftungen des Events aus dem Signal herausgezogen.

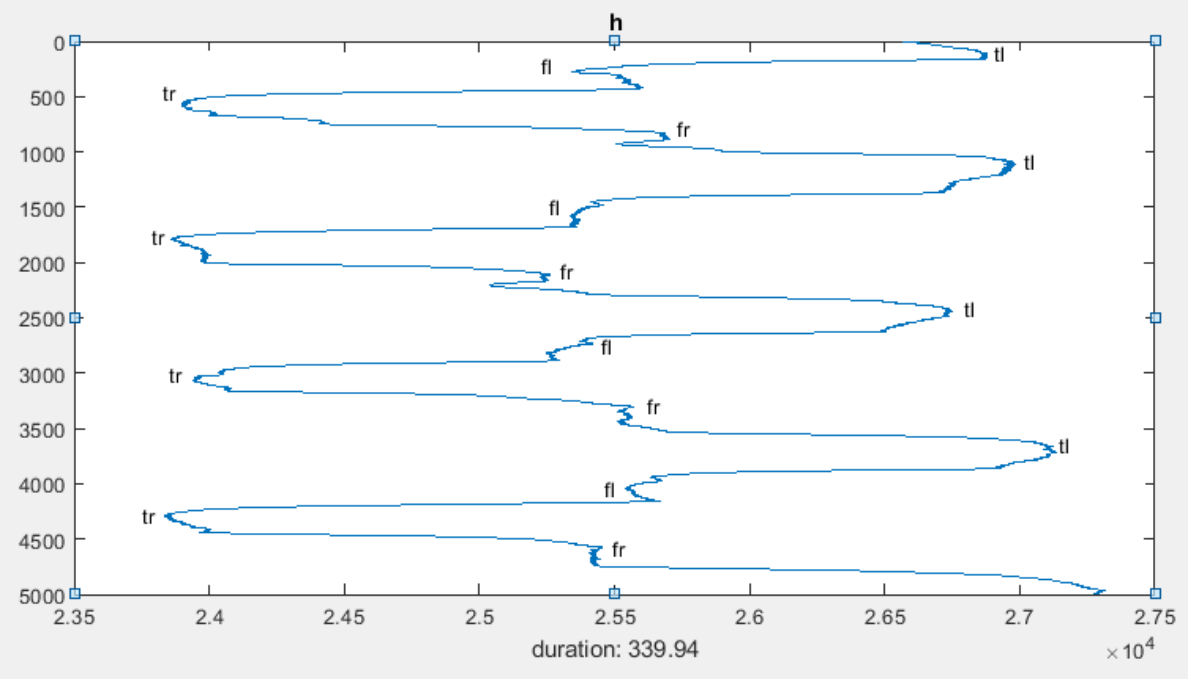


Abbildung 7-19: „online“ Auswertung für horizontales EOG Signal mit erkannten Events

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Auch für die „online“ Auswertung der Events ist die Kalibrierung entscheidend. Damit die Werte nicht jedes Mal neu bestimmt werden müssen, wurde für die Tests mithilfe des Skriptes „liveeog_getreferencecalibration.m“ (siehe Anhang C) nach 8000 Messpunkten eine automatische Kalibrierung durchgeführt. Diese Funktion bestimmt die verschiedenen Kalibrierungsparameter anhand der letzten 5000 Messpunkte.

7.3.4 Portieren des Algorithmus

Da die Matlab-Scrips strukturiert sind, können sie direkt für das Erstellen einer DLL mit Hilfe des Matlab Compilers [51] verwendet werden. Die Funktion liveeog(v,h,freq,init) bildet dabei die Schnittstelle nach außen. Als Parameter muss das vertikale und das horizontale EOG Signal übergeben werden sowie die Abtastfrequenz. Um alle Speicher zu initialisieren muss bei dem ersten Aufruf der Funktion der Parameter „init“ gesetzt sein. Um eine DLL von Matlab verwenden zu können, muss die kostenlose Runtime-Umgebung von Matlab installiert sein. Da bis zum Abschluss dieser Arbeit keine Runtime von Maltlab für Android zur Verfügung stand, konnte dieser letzte Schritt nicht durchgeführt werden. Auf die Möglichkeit der entfernten Auswertung auf einem im Internet befindlichen Server wurde aufgrund der zu erwartenden Verzögerung bei der Auswertung des Signals nicht zurückgegriffen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

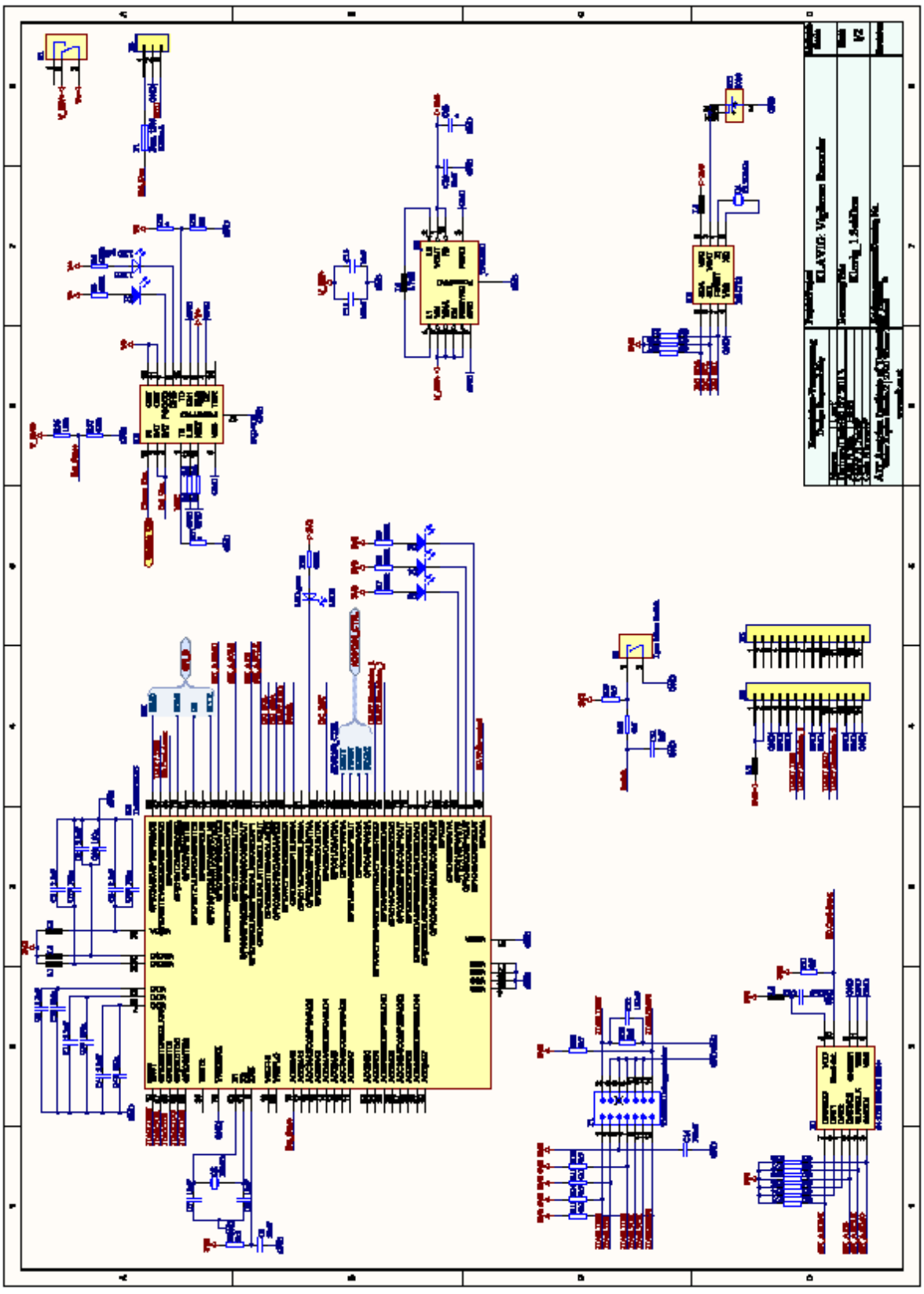
8. Zusammenfassung und Ausblick

Durch das Einbringen von neuen Technologien lassen sich Biosignalerfasser mittlerweile sehr klein und kompakt bauen. Durch die Verwendung von Smartphones oder Tablets in Verbindung mit schnellen drahtlosen Übertragungstechnologien lassen sich Anzeigen und Steuerung der Messgeräte, unabhängig von Größe und Bauform des eigentlichen Messgerätes, verwirklichen. Bezogen auf die umgesetzten Erweiterungen des Biosignalerfassers aus dem KlaVig Projekt konnten schon die oben genannten Vorteile in anderen Projekten und Studien in der Praxis gezeigt werden.
Durch die Einstellbarkeit der Parameter des Biosignalerfassers, ohne die Firmware neu zu kompilieren, können nun die Messparmeter leichter und schneller an die Bedingungen angepasst werden. Auch die geforderte regelmäße Korrektur der RTC durch die Android Applikation erleichtert das zuordnen der Messungen. Der in dieser Arbeit realisierte Echtzeit-Signalverlauf in der Android Applikation erleichtert das Messen von Biosignalen, da nun der Signalverlauf einfach kontrolliert werden kann, bevor die Messung gestartet wird, und nicht erst bei der Auswertung der Aufzeichnung. Auch während einer Messung kann der Signalverlauf kontrolliert werden und gegebenfalls die Messung abgebrochen und neu gestartet werden, oder wiederholt werden. Die Umsetzung dieser Anforderung ermöglich eine raschere und zuverlässigere Aufzeichnung nicht nur von EOG-Signalen sondern aller Biosignale die mit diesem Biosignalerfasser gemessen werden können.
Der Auswertealgorithmus liefert vergleichbare Ergebnisse wie bei den bereits bekannten Arbeiten und zitierten Arbeiten. So werden die klassischen Augenpositionen „oben“, “unten“, „links“, „rechts“ und „Mitte“ erkannt. Dadurch, dass die Auswertung mithilfe des Dipol-Modelles arbeitet und nicht ein Signalmuster erkennt, wie in anderen Arbeiten, wird eine zusätzliche Möglichkeit geschaffen Steuersignale aus dem EOG-Signal abzuleiten. Die detaillierte Auswertung der Augenbewegung ermöglicht es sogenannte Events zu erkennen. Diese Events werden bei Augenbewegungen zu einer der bekannten Augenpositionen generiert, wenn die Augenbewegung schnell genug erfolgt. Sowohl die Augenposition als auch die Events werden durch den Algorithmus zuverlässig erkannt. Dies gilt sowohl für Auswertung von aufgezeichneten Signalen im Nachhinein als auch für eine Auswertung in Echtzeit. Durch die Glättung des Signals kann mit Hilfe des XY Darstellung auch die Blickrichtung ermittelt werden.
Entscheidend für die korrekte Erkennung der Zustände und Events ist, wie in anderen Arbeiten auch, die Kalibrierung des Systems. Eine automatische Kalibrierung des Systems ist der nächste logische Schritt, der notwendig ist, um einen zuverlässigen Prototypen für ein EOG basierendes Steuergerät zu entwickeln. Dabei muss zum Teil auf Änderungen im Signal geachtet werden deren Ursprung durch Änderungen in der Umgebung wie z.B.: Lichtverhältnisse oder durch das Verhalten des/r Nutzers/ins wie z.B. Lesen dieselben Charakteristiken haben. Im Fall von geänderten Lichtverhältnissen muss die Kalibrierung verändert werden, im Falle von Lesen muss sie nicht geändert werden, da dazu keine extremen Augenposition verwendet werden.
Wie diese Arbeit zeigt funktioniert der Algorithmus unter Laborbedingungen, da hier eine neue Kalibrierung vor dem Test erfolgen kann.

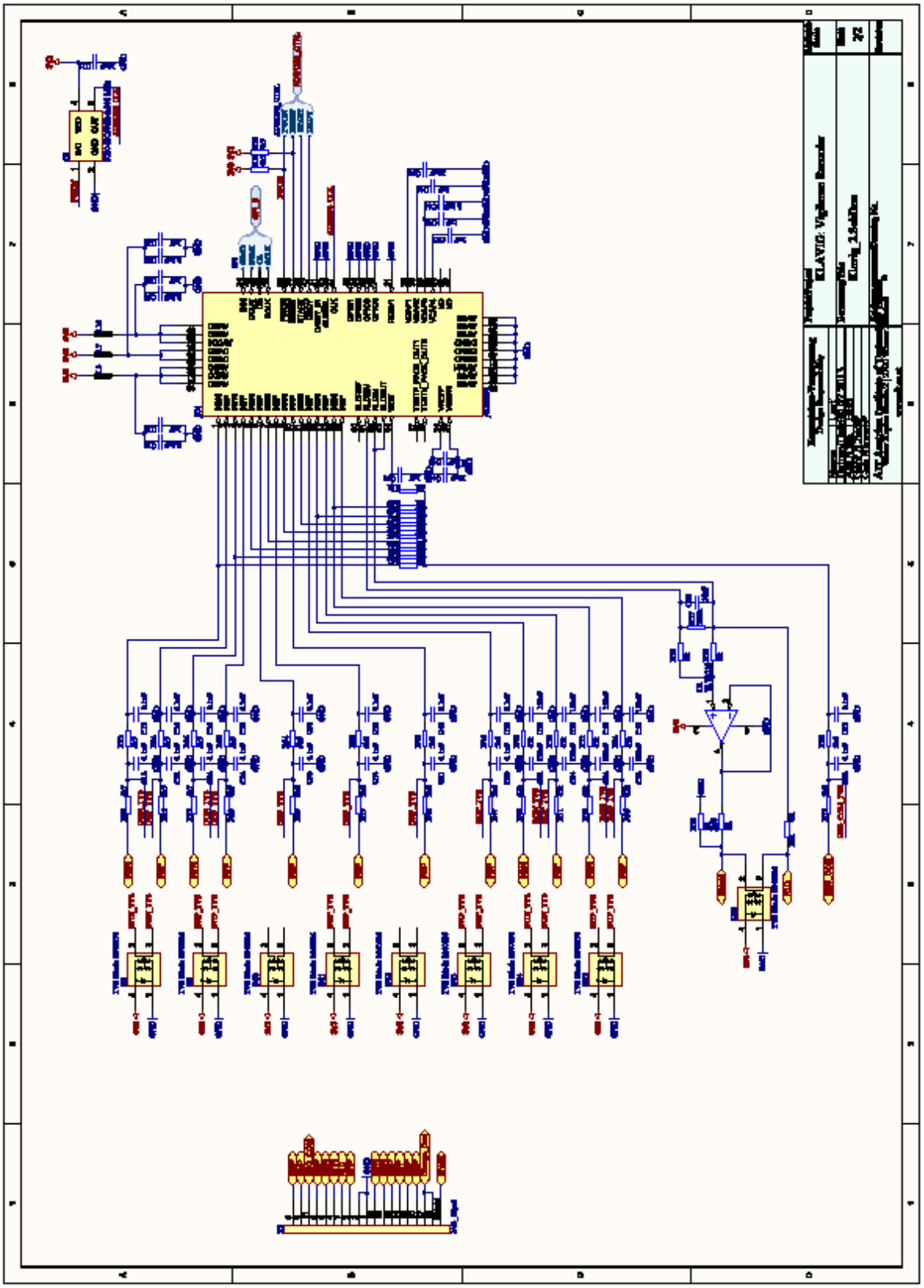
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Anhang A

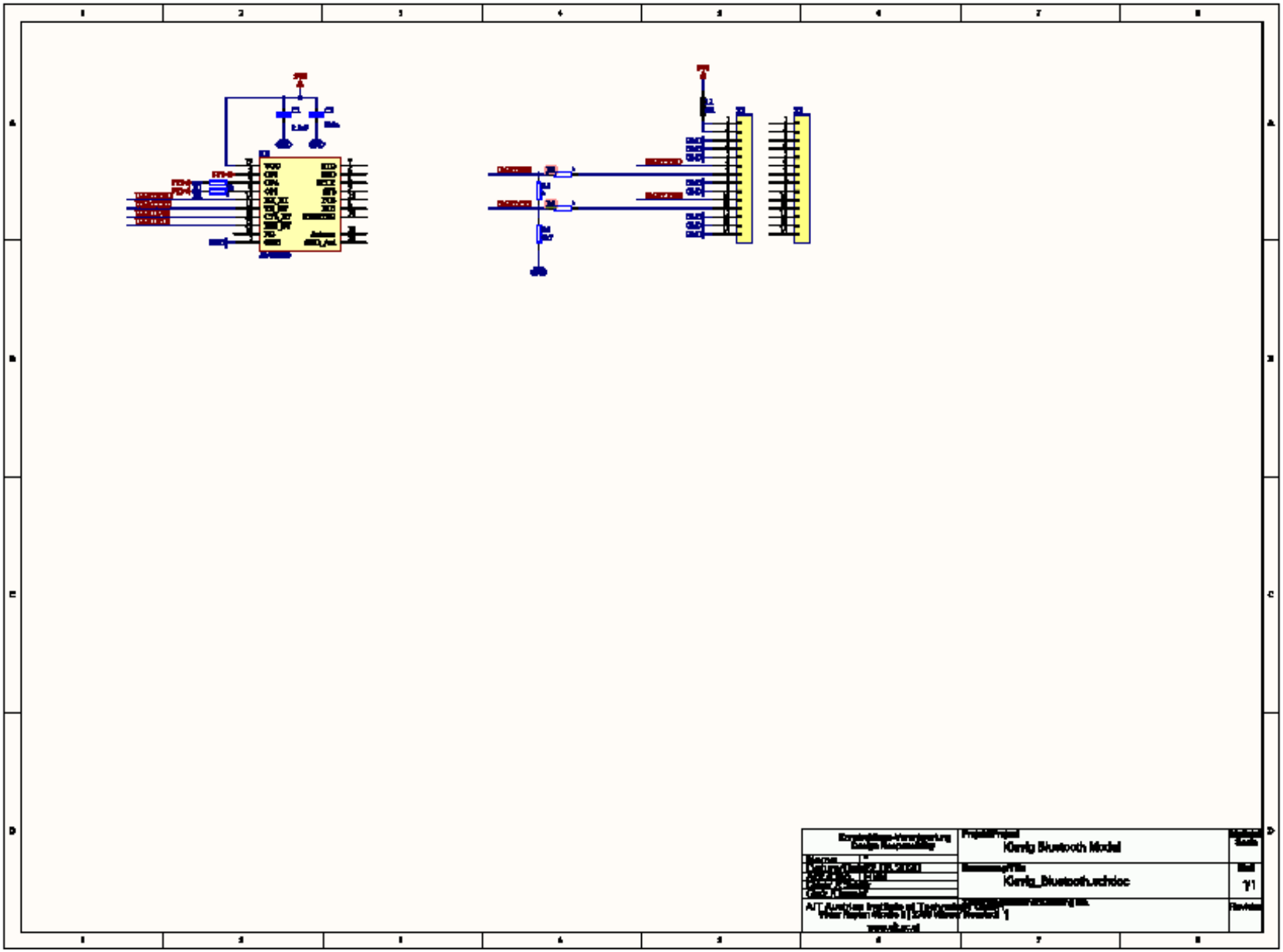


TU Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.



TU Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.



Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Anhang B

```
%#eml
% liveeog
% author: Wolfgang Wild AIT Austrian Institute of Technology GmbH
% version 0.1
% Auswertung von EOG Signalen
% INPUT:
% v ... neuer vertikaler wert
% h ... neuer horizontaler wert
% freq ... abtastrate
%
% OUTPUT:
% ann ... annotationen
% ve ... vertikales eventsignal
% he ... horizontales eventsignal
% x ... x-mauskoordinate
% y ... y-mauskoordinate
% vg ... vertikales geglättetes signal
% hg ... horizontales geglättetes signal

function [ann, ve, he, x, y, vg, hg] = liveeog(v, h, freq, init)

global calibdata
persistent thresholds
persistent liveeogdata

ann = '';
ve = v;
he = h;
x = 0;
y = 0;
vg = 0;
hg = 0;

einschwingdauer = round(freq/4);

% Falls keine Kalibrierdaten vorher geladen wurden, werden Referenzwerte
% verwendet
if isempty(calibdata)
    calibdata = liveeog getreferencecalibration();
end

% Thresholds berechnen
thresholds = liveeog thresholds(calibdata);

% Beim ersten Aufruf bzw. bei einem Reset alles löschen
if isempty(liveeogdata) || init == 1
   liveeogdata.v = zeros(8*freq, 1);
   liveeogdata.h = zeros(8*freq, 1);
   liveeogdata.einschwingen.ignore = einschwingdauer;
   liveeogdata.einschwingen.grundwert = einschwingdauer;
   liveeogdata.grundwert.v = 0;
   liveeogdata.grundwert.h = 0;
   liveeogdata.expmean.v = 0;
   liveeogdata.expmean.h = 0;
end

% Einschwingen von Filtern, Werte ignorieren
if liveeogdata.einschwingen.ignore
    liveeogdata.einschwingen.ignore = liveeogdata.einschwingen.ignore - 1;
    return
end

% Nach dem Einschwinge Grundwerte ermitteln
if liveeogdata.einschwingen.grundwert
    liveeogdata.einschwingen.grundwert = liveeogdata.einschwingen.grundwert - 1;
    liveeogdata.grundwert.v = liveeogdata.grundwert.v + v;
    liveeogdata.grundwert.h = liveeogdata.grundwert.h + h;

    if liveeogdata.einschwingen.grundwert == 0
        liveeogdata.grundwert.v = liveeogdata.grundwert.v / einschwingdauer;
        liveeogdata.grundwert.h = liveeogdata.grundwert.h / einschwingdauer;

        liveeogdata.v = repmat(liveeogdata.grundwert.v, size(liveeogdata.v));
```


Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
74              liveeogdata.h = repmat(liveeogdata.grundwert.h, size(liveeogdata.h));
75          end
76          return
77      end
78
79      % buffer shift
80      liveeogdata.v = [liveeogdata.v(2:end); v];
81      liveeogdata.h = [liveeogdata.h(2:end); h];
82
83
84      % zwinkern entfernen und signal glätten
85      windowsize = round(freq/2);
86      vg = median(liveeogdata.v(end-windowsize:end));
87      hg = median(liveeogdata.h(end-windowsize:end));
88
89
90      % mauskoordinaten berechnen
91      % ToDo
92      y = vg - liveeogdata.grundwert.v;
93      x = hg - liveeogdata.grundwert.h;
94
95
96      % events erkennen und übersetzen
97      ve = vg - median(liveeogdata.v);
98      he = hg - median(liveeogdata.h);
99      event = liveeog_events(ve, he, liveeogdata.v(end-round(windowsize/2)),
        liveeogdata.h(end-round(windowsize/2)), thresholds, freq);
100
101     switch event.v
102         case 2 % to top
103             ann = 'tt';
104         case 1 % from bottom
105             ann = 'fb';
106         case -1 % from top
107             ann = 'ft';
108         case -2 % to bottom
109             ann = 'tb';
110     end
111
112     switch event.h
113         case 2 % to left
114             ann = [ann 'tl'];
115         case 1 % from right
116             ann = [ann 'fr'];
117         case -1 % from left
118             ann = [ann 'fl'];
119         case -2 % to right
120             ann = [ann 'tr'];
121     end
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

```
%#eml
% liveeog events
% author: Wolfgang wild AIT Austrian Institute of Technology GmbH
% version 0.1
% INPUT:
% v ... letzter vertikaler wert, driftbereinigt
% h ... letzter horizontaler wert, driftbereinigt
% v_orig ... letzter vertikaler wert, original
% h_orig ... letzter horizontaler wert, original
% threshholds ... grenzwerte aus calibration
% freq ... abtastrate
%
% OUTPUT:
% event ... numerische Repräsentation von gefunden Events für h und v

function [event] = liveeog_events(v, h, v_orig, h_orig, thresholds, freq)

persistent past
persistent state

% init
if isempty(state)
    state.v = 0;
    state.h = 0;
end

if isempty(past)
    past.v = v_orig;
    past.h = h_orig;
    past.dv = zeros(round(0.3 * freq), 1);
    past.dh = zeros(round(0.3 * freq), 1);
end

% die letzten werte der ersten ableitung buffern
past.dv = [past.dv(2:end); v_orig-past.v];
past.dh = [past.dh(2:end); h_orig-past.h];

% statemachine und eventerkennung
[state.v, event.v] = onedimension_states(v, past.dv, thresholds.v, state.v);
[state.h, event.h] = onedimension_states(h, past.dh, thresholds.h, state.h);

% aufraeumen
past.v = v_orig;
past.h = h_orig;



%% zustandsänderungen und events auslösen
function [state, event] = onedimension_states(s, d, t, state)

event = 0;

% zustandsänderung: ganz oben bzw. rechts
if s > t.pos.far && state ~= 1
    state = 1;

    % rasche zustandsänderung: event
    if max(d) > t.diff.upper.up
        event = 2;
    end


% zustandsänderung: ganz unten bzw. links
elseif s < t.neg.far && state ~= -1
    state = -1;

    % rasche zustandsänderung: event
    if min(d) < t.diff.lower.down
        event = -2;
    end

% zustandsänderung: von oben bzw. rechts zur mitte
elseif s < t.pos.near && s > t.neg.near && state == 1
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
74          state = 0;
75
76          % rasche zustandsänderung: event
77          if min(d) < t.diff.upper.down
78              event = -1;
79          end
80
81      % zustandsänderung: von unten bzw. links zur mitte
82      elseif s < t.pos.near && s > t.neg.near && state == -1
83          state = 0;
84
85          % rasche zustandsänderung: event
86          if max(d) > t.diff.lower.up
87              event = 1;
88          end
89      end
90
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

```
%#eml
% liveeog generateReference
% author: Wolfgang Wild AIT Austrian Institute of Technology GmbH
% version 0.1
% durchsucht das Signal und erstellt Referenzwerte
% INPUT:
% data ... Signaldaten (2 Kanäle)
% f ... abtastfrequenz

function liveeog_generateReference(data, freq)
global calibdata

h = data(:,3);
v = data(:,2);

% Einschwingdauer wie in liveeog
einschwingdauer = round(freq/4);

% Die ersten 2 Einschwingdauern ignorieren_

% Daten ab der 3_ Einschwingdauer benutzen
v = v(2*einschwingdauer+1:end);
h = h(2*einschwingdauer+1:end);

% zwinkern entfernen und signal glätten
windowsize = round(freq/2);
vg = movmedian(v, windowsize);
hg = movmedian(h, windowsize);

% medianfilter wie in liveeog vor der Eventerkennung (6 sekunden fenster)
ve = vg - movmedian(v, 6*freq);
he = hg - movmedian(h, 6*freq);

% Erste Ableitung wie in liveeog_events
dv = diff(v);
dh = diff(h);

%% Referenzwerte ermitteln
% Nimm nicht Max und Min, sondern das 5% oder 10% und das 90% oder 95%
% Perzentil, um Ausreißer zu ignorieren
% Strichpunkte und punkte fehlen absichtlich, um die Werte auszugeben

format compact

fuenfneunzig = round(95 *length(ve) / 100);
neunzig = round(90 *length(ve) / 100);
zehn = round(10 *length(ve) / 100);
fuenf = round(5 *length(ve) / 100);

ve = sort(ve);
dv = sort(dv);
he = sort(he);
dh = sort(dh);

calibdata_v_abs_top         = ve(neunzig)       % Vertikales Maximum = Top
calibdata_v_abs_bottom      = ve(zehn)          % Vertikales Minimum = Bottom
calibdata_v_screen_top      = ve(neunzig)/10    % Mittenbereich ("Oberer
Bildschirmrand")
calibdata_v_screen_bottom   = ve(zehn)/10       % Mittenbereich ("Unterer
Bildschirmrand")
calibdata_v_diff_lower_up   = dv(fuenfneunzig)  % Maximum 1_ Ableitung = Event nach
oben
calibdata_v_diff_lower_down = dv(fuenf)         % Minimum 1_ Ableitung = Event nach
unten
calibdata_v_diff_upper_up   = dv(fuenfneunzig)  % Maximum 1_ Ableitung = Event nach
oben
calibdata_v_diff_upper_down = dv(fuenf)         % Minimum 1_ Ableitung = Event nach
unten


calibdata_h_abs_left        = he(neunzig)       % Horizontales Maximum = Left
calibdata_h_abs_right       = he(zehn)          % Horizontales Minimum = Right
calibdata_h_screen_left     = he(neunzig)/10    % Mittenbereich ("Linker
```


TU Bibliothek Your knowledge hub WIEN
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

```
    Bildschirmrand")
68  calibdata h screen right    = he(zehn)/10       % Mittenbereich ("Rechter
    Bildschirmrand")
69  calibdata h diff lower up   = dh(fuenfneunzig)  % Maximum 1  Ableitung = Event nach
    links
70  calibdata h diff lower down = dh(fuenf)         % Minimum 1  Ableitung = Event nach
    rechts
71  calibdata_h_diff_upper_up   = dh(fuenfneunzig)  % Maximum 1_ Ableitung = Event nach
    links
72  calibdata_h_diff_upper_down = dh(fuenf)         % Minimum 1_ Ableitung = Event nach
    rechts
73
74  format loose
75
76  % Schreibe Werte ins globale Struct
77  calibdata.v.abs.top          = calibdata_v_abs_top;
78  calibdata.v.abs.bottom       = calibdata_v_abs_bottom;
79  calibdata.v.screen.top       = calibdata_v_screen_top;
80  calibdata.v.screen.bottom    = calibdata_v_screen_bottom;
81  calibdata.v.diff.lower.up    = calibdata_v_diff_lower_up;
82  calibdata.v.diff.lower.down = calibdata_v_diff_lower_down;
83  calibdata.v.diff.upper.up    = calibdata_v_diff_upper_up;
84  calibdata.v.diff.upper.down = calibdata_v_diff_upper_down;
85
86  calibdata.h.abs.left         = calibdata_h_abs_left;
87  calibdata.h.abs.right        = calibdata_h_abs_right;
88  calibdata.h.screen.left      = calibdata_h_screen_left;
89  calibdata.h.screen.right     = calibdata_h_screen_right;
90  calibdata.h.diff.lower.up    = calibdata_h_diff_lower_up;
91  calibdata.h.diff.lower.down = calibdata_h_diff_lower_down;
92  calibdata.h.diff.upper.up    = calibdata_h_diff_upper_up;
93  calibdata.h.diff.upper.down = calibdata_h_diff_upper_down;
94
95
```

```
 1  %#eml
 2  % liveeog thresholds
 3  % author: Wolfgang Wild AIT Austrian Institute of Technology GmbH
 4  % version 0.1
 5  % Grenzwerte aus Kalibrierdaten berechnen
 6  % INPUT:
 7  % calibdata ... Kalibrierungsdaten
 8
 9  function [thresholds] = liveeog_thresholds(calibdata)
10
11  thresholds.v.pos.near           = calibdata.v.screen.top + (calibdata.v.abs.top -
    calibdata.v.screen.top) / 3;
12  thresholds.v.pos.far            = calibdata.v.screen.top + 2 * (calibdata.v.abs.top
    - calibdata.v.screen.top) / 3;
13  thresholds.v.neg.near           = calibdata.v.screen.bottom -
    (calibdata.v.screen.bottom - calibdata.v.abs.bottom) / 3;
14  thresholds.v.neg.far            = calibdata.v.screen.bottom - 2 *
    (calibdata.v.screen.bottom - calibdata.v.abs.bottom) / 3;
15  thresholds.v.diff.lower.up      = calibdata.v.diff.lower.up / 2;
16  thresholds.v.diff.lower.down    = calibdata.v.diff.lower.down / 2;
17  thresholds.v.diff.upper.up      = calibdata.v.diff.upper.up / 2;
18  thresholds.v.diff.upper.down    = calibdata.v.diff.upper.down / 2;
19
20  thresholds.h.pos.near           = calibdata.h.screen.left + (calibdata.h.abs.left -
    calibdata.h.screen.left) / 3;
21  thresholds.h.pos.far            = calibdata.h.screen.left + 2 *
    (calibdata.h.abs.left - calibdata.h.screen.left) / 3;
22  thresholds.h.neg.near           = calibdata.h.screen.right -
    (calibdata.h.screen.right - calibdata.h.abs.right) / 3;
23  thresholds.h.neg.far            = calibdata.h.screen.right - 2 *
    (calibdata.h.screen.right - calibdata.h.abs.right) / 3;
24  thresholds.h.diff.lower.up      = calibdata.h.diff.lower.up / 2;
25  thresholds.h.diff.lower.down    = calibdata.h.diff.lower.down / 2;
26  thresholds.h.diff.upper.up      = calibdata.h.diff.upper.up / 2;
27  thresholds.h.diff.upper.down    = calibdata.h.diff.upper.down / 2;
28
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
%#eml
% liveeog offline forKlavig
% author: Wolfgang Wild AIT Austrian Institute of Technology GmbH
% version 0.1
%% Offline-Ausfürhung von liveeog zu Testzwecken
% INPUT:
% data ... Daten von 2 Kanälen
% freq ... abtastfrequenz
% liveview ... boolean live auswertung=true; nur Endergebnis=false

function [] = liveeog_offline_forKlavig(data, freq, liveview)
global calibdata;

if (nargin < 2)
    freq = 256;
end

if (nargin < 3)
    liveview = true;
end

%% Um Kalibrierdaten aus dem einen Datensatz zu bestimmen:
liveeog_generateReference(data, freq)

%% Um Kalibrierdaten aus liveeog_getreferencecalibration zu laden:
%calibdata = liveeog_getreferencecalibration()


h = data(:,2);
v = data(:,3);

t = (1:length(v))./freq;

ann = cell(length(data), 1);
vm = zeros(length(data), 1);
hm = zeros(length(data), 1);
vg = zeros(length(data), 1);
hg = zeros(length(data), 1);
x = zeros(length(data), 1);
y = zeros(length(data), 1);


[ann{1}, vm(1), hm(1), x(1), y(1), vg(1), hg(1)] = liveeog(v(1), h(1), freq, 1);

disp([8 ' ... 000%']);


tic % time real
for i = 2:length(v)
    t_sim = i/freq;


    [ann{i}, vm(i), hm(i), x(i), y(i), vg(i), hg(i)] = liveeog(v(i), h(i), freq, 0);

    if mod(i,floor(length(v)/100)) == 0
        disp([8 8 8 8 8 num2str(ceil(100*i/length(v)), '%03d%%')])
    end

    if (toc < t_sim) && liveview
        a = max(1, i-5*freq);

        subplot(221)
        plot(v(a:i));
        grid on
        title('v');
        for j = a:i
            if ~isempty(ann{j}) && (ann{j}(2) == 't' || ann{j}(2) == 'b')
                text(j-a, v(j), ann{j});
            end
        end
        xlabel(t sim);

        subplot(223)
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

```
            plot(-h(a:i));
            title('h')
            for j = a:i
                if ~isempty(ann{j}) && (ann{j}(2) == 'l' || ann{j}(2) == 'r')
                    text(j-a, -h(j), ann{j});
                end
            end
            view(90,90);

            subplot(122)
            plot(-h(a:i), v(a:i));
            title('xy');
            for j = a:i
                if ~isempty(ann{j})
                    text(-h(j), v(j), ann{j});
                end
            end

            drawnow
        end
    end

    t = t(3*freq:end);
    v = v(3*freq:end);
    h = h(3*freq:end);
    vm = vm(3*freq:end);
    hm = hm(3*freq:end);
    vg = vg(3*freq:end);
    hg = hg(3*freq:end);
    x = x(3*freq:end);
    y = y(3*freq:end);
    ann = ann(3*freq:end);

    subplot(221)
    plot(t, v);
    hold all
    plot(t, vg, 'LineWidth', 3);
    plot(t, vm, 'LineWidth', 3);
    plot(t(2:end), diff(v));
    title('v');
    for i = 1:length(ann)
        if ~isempty(ann{i}) && (ann{i}(2) == 't' || ann{i}(2) == 'b')
       %     text(t(i), 0, ann{i});
            xline(t(i));
        end
    end
    hold off

    subplot(223)
    plot(t, h);
    hold all
    plot(t, hg, 'LineWidth', 3);
    plot(t, hm, 'LineWidth', 3);
    plot(t(2:end), diff(h));
    title('h')
    for i = 1:length(ann)
        if ~isempty(ann{i}) && (ann{i}(2) == 'l' || ann{i}(2) == 'r')
          %  text(t(i), 0, ann{i});
            xline(t(i));
        end
    end
    hold off

    subplot(122)
    plot(-h, v);
    hold all
    plot(-hg, vg, 'LineWidth', 3);
    plot(-hm, vm, 'LineWidth', 3);
    plot(diff(h), diff(v));
    legend('Originalsignal', 'geglaettetes Signal', 'Zustandserkennung', 'diff\_orig'
    'Location', 'NorthWestOutside');
    title('xy');
    hold off
```


Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
%#eml
% liveeog online forKlavig
% author: Wolfgang Wild AIT Austrian Institute of Technology GmbH
% version 0.1
%% Online-Ausfürhung von liveeog zu Testzwecken
% INPUT:
% Output:

%% Online-Ausführung von liveeog über serielle Schnittstelle
function [] = liveeog_online()

global KEY_IS_PRESSED
global AXIS_LOCKED
global axis_values
global s

KEY_IS_PRESSED = 0;
AXIS_LOCKED = 0;

freq = 250;

% beende alles was übergelassen wurde
%instrumentObjects=instrfind
%delete(instrumentObjects)

disp('defining serial port parameters')
s = serial('COM17','BaudRate',115200);          % Objekt auf Serial Port Anlegen
s.InputBufferSize = 15500;

disp('opening connection (this may take some time...)')
fopen(s);

% Figure with FunctionHandle in case of KeyPress
gcf
set(gcf, 'KeyPressFcn', @myKeyPressFcn);
drawnow

disp('start')
fwrite(s, ['stream' 13 10]);
liveeog(0, 0, freq, 1);

mybuffer = [];
eog = NaN(6000,2);
maus = zeros(6000,2);
anns = cell(6000, 1);
refcount = 0;
calibrate = true;

tic
% try
    while ~KEY_IS_PRESSED

        % collect data
        if s.BytesAvailable
            currentdata = fread(s, s.BytesAvailable, 'uint8');  % read all you've
            got for me
            mybuffer = [mybuffer; currentdata];                 % and write to my
            own buffer, i'll deal with it later
    %         length(mybuffer)                                    % write buffer
    lenght to console
        end


        % parse data
        % Dec Wert 13 ist carriage return (\r)
        while(ismember(13, mybuffer))        % have you got more than one entire
        data frame for me?
            frameende = find(mybuffer==13, 1);        % Position von \r im Datenstrom
            data = mybuffer(1:frameende-1);          % read entire data frame
            mybuffer = mybuffer(frameende+2:end);    % shift buffer by the lengh of
            one frame

            % Daten zerlegen:
```


Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
69              data_kanaele = strsplit(char(data'), ';');
70              int_kanaele = str2double(data_kanaele);
71              refcount = refcount + 1;
72
73              % do the liveEOG calculations
74              [ann, ve, he, xl, yl, vg, hg] = liveeog(int_kanaele(3), int_kanaele(2),
                freq, 0);
75
76              % buffer the things we want to see
77              eog = [eog(2:end,:); [int_kanaele(3), int_kanaele(2)]];
78              maus = [maus(2:end,:); [maus(end,1)+ve maus(end,2)+he]];
79              anns = [anns(2:end); {ann}];
80          end
81
82          if refcount > 8000 && calibrate
83              liveeog_generateReference([zeros(6000,1), eog], freq);
84              calibrate = false;
85          end
86
87
88          % Filter
89          % 50hz Filter
90          [B,A] = butter(4, 30/(freq/2), 'low');
91          vsmooth = filter(B,A,double(eog(:,2)));
92          hsmooth = filter(B,A,double(eog(:,1)));
93          vsmooth = vsmooth(1001:end);
94          hsmooth = hsmooth(1001:end);
95          annsplot = anns(1001:end);
96
97          % plot when you find some time to do so...
98          t = toc;
99          tic
100
101         % eog horizontal
102         subplot(222)
103         plot(hsmooth);
104         view(90,90);
105         title('h')
106         ylabel(['duration: ' num2str(refcount/freq)]);
107         for j = 1:8000
108             if ~isempty(annsplot{j}) && (annsplot{j}(2) == 'r' || annsplot{j}(2) ==
                'l')
109                 text(j, hsmooth(j), annsplot{j});
110             end
111         end
112
113         % eog vertikal
114         subplot(223)
115         plot(vsmooth);
116         title('v');
117         for j = 1:8000
118             if ~isempty(annsplot{j}) && (annsplot{j}(2) == 't' || annsplot{j}(2) ==
                'b')
119                 text(j, hsmooth(j), annsplot{j});
120             end
121         end
122
123         % eog xy
124         subplot(224)
125         hold off
126         plot(hsmooth, vsmooth);
127         hold on
128         plot(hsmooth(end), vsmooth(end), 'r*');
129         if AXIS_LOCKED
130             axis(axis_values);
131         end
132         title('xy');
133         xlabel(['drawing frame rate: ' num2str(1/t)]);
134
135         drawnow
136
137     end
138 % catch
```


TU Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

```
139 %     disp('closing connection');
140 %     fclose(s);
141 % end
142
143 disp('stop');
144 fwrite(s, 'stop\r\n')
145
146 disp('closing connection');
147 fclose(s);
148
149 end
150
151
152
153 %% Hand over this function to the plotting window as FunctionHandle
154 % "hObject" tells us in which window the keypress occured. We don't care
155 % for that, but we need it to get the type signature right.
156 % "event" contains information about what happened. We will react
157 % accordingly.
158 function myKeyPressFcn(hObject, event)
159     global KEY_IS_PRESSED
160     global AXIS_LOCKED
161     global axis_values
162
163     % if 'x' is pressed, tell the above loop to terminate
164     if event.Key == 'x'
165         KEY_IS_PRESSED  = 1;
166         disp('x key is pressed')
167
168     % if 'l' is pressed, toggle the axes lock for plotting
169     elseif event.Key == 'l'
170         if AXIS_LOCKED
171             disp('unlocking axes')
172             AXIS_LOCKED = 0;
173         else
174             disp('locking axes')
175             AXIS_LOCKED = 1;
176         end
177         axis_values = axis;
178     end
179 end
180
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Abbildungsverzeichnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Tabellenverzeichnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Literaturverzeichnis

[1] ISO, *ISO 13485:2016,* Genf, Schweiz: ISO, 2016.

[2] W. Dröschel, Das V - Modell 97, München, Wien, Oldenbourg: degruyter, 2000.

[3] K. Pohl und C. Rupp, Basiswissen Requirements Engineering, Heidelberg: dpunkt.verlag, 2015.

[4] W. Kahle, H. Leonhardt und W. Platzer, Taschenatlas der Anatomie 3 Nervensystem und Sinnesorgane, Stuttgart: Georg Thieme, 1986.

[5] C. Fahlke, W. Linke, B. Rassler und W. R., Taschenatlas Physiologie, München: Urban & Fischer, 2015.

[6] H. H., Biomedizinische Technik Bd.1 Diagnostik und bildgebende, Berlin: Springer, 1992.

[7] A. Bethe und H. Kast, Synergische und reziproke Innervation antagonistischer Muskeln nach Versuchen am Menschen nebst Beobachtungen über ihre Reaktionszeit, Springer, 1922.

[8] E. Hering, Die Lehre vom binocularen Sehen., Leipzig: W. Engelmann, 1868.

[9] N. J. Buchner Helmut, Evozierte Potenziale, neurovegetative, Diagnostik, Okulographie: methodik und klinische Anwendungen, Stuttgart: Georg Thieme Verlag, 2005.

[10] S. Hanke, *EOG-gesteuerte Rechnersysteme mit Body-Area-Network-Anbindung,* Dresden: Technische Universität Dresden, 2004.

[11] J. G. Webster, Medical Instrumentation Application and Design, Canada: Hoboken, NJ : Wiley, 2010.

[12] S. Brunner, *Entwicklung, Implementierung und Validierung der Signalverarbeitung für ein EOG-Kummunikations- und Steuerungssystem,* Linz: Medizintechnik Linz, 2007.

[13] S. Hanke, *Realtime Assessment of Vigilance based on integrated EOG biosignal pattern recognition,* Wien: Medizinische Universität Wien, 2012.

[14] P. Trillenberg, „Elektrookulographie," *Das Neurophysiologie-Labor,* Bd. 34, Nr. 34, pp. 98-106, 02 08 2012.

[15] A. Bulling, J. A. Ward, H. Gellersen und T. Gerhard, „Eye Movement Analysis for Activity Recognition Using Electrooculography," *IEEE Transaction on Pattern Analysis and Machine intelligenca,* pp. 741-753, Nr.: 4 VOL.33 2011.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[16] A. Bulling, J. A. Ward, H. Gellersen und G. Tröster, „Robust Recognition of Reading Activity in Transit Using Wearable Electrooculography," *Lecture Botes in Computer Science book* , pp. 19-37, Volume 5013 2008.

[17] G. B. Arden und P. A. Constable, „The electro-oculogram," *Retinal and Eye Research* , pp. 207-248, Nr.25 2006.

[18] E. Iánez, A. Úbeda und P.-V. Carlos, „Assistive robot application based on an RFID control architecture and a wireless EOG interface," *Robotics and Autonomous Systems 60,* pp. 1069 - 1077, 2012.

[19] C.-C. Postelnicu, G. Florin und T. Doru, „EOG-based visual navigation interface development," *Expert Systems with Applications 39,* pp. 10857-10866, 2012.

[20] A. R. Kherlopian, J. P. Gerrein, M. Yue und K. E. Kim, „Electrooculogram based system for computer control using a multiple feature classification model," in *EMBS Annual International Conference* , New Yourk City, USA, Aug 30 - Sept 03, 2006.

[21] Z. Lv, X.-p. Wu, L. Mi und Z. Dexiang, „A novel eye movement detection algorithm for EOG driven human coputer interface," *Pattern Recognition Letters 31,* pp. 1041-1047, 2010.

[22] T. Künneth, Android 4 Apps entwickeln mit dem Android SDK, Bonn: Galileo Press, 2012.

[23] V. K. Ingle und J. G. Proakis, Digital signal processing using MATLAB V.4, Boston: PWS Publishing Company, 1997.

[24] U. Zimmermann und O. Harald, Messtechnik für Ingenieure und Praktiker, Aachen: Shaker, 2009.

[25] E. Wintermantel und S.-W. Ha, Medizintechnik Life Science Engineering, Berlin: Springer, 2009.

[26] G. Fasching, Werkstoffe für die Elktrotechnik 3. Auflage, Wien: Springer, 1994.

[27] J. Eichmeier, Medizinische Elektronik: eine Einführung 3. Auflage, Berlin, Heidelberg, New York: Springer, 1997.

[28] P. Husar, Biosignalverarbeitung, Berlin Heidelberg: Springer, 2010.

[29] R. Lerch, Elektrische Messtechnik : Analoge, digitale und computergestützte Verfahren, Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg New York, 2005.

[30] F. Puente León und U. Kiencke, Messtechnik: Systemtheorie für Ingenieure und Informatiker, Berlin, Heidelberg: Springer Berlin Heidelberg 2012, 2012.

[31] J. Kurose und K. Ross, Computernetzwerke, München: Pearson , 2012.

[32] A. S. Tannenbaum, Computernetzwerke, Münschen: Prentice Hall, 1998.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[33] K. Obermann und H. Martin, Datennetztechnologien für next Generation Networks, Wiesbaden: Springer Vieweg+Teubner, 2009.

[34] Bluethooth Special Interest Group, „www.Bluetooth.com," 24 07 2012. [Online]. Available: https://www.bluetooth.org/docman/handlers/DownloadDoc.ashx?doc_id=260866&vId=290097. [Zugriff am 21 6 2019].

[35] A. S. Huang, Bluetooth Essentials for Programmers, Cambridge : Cambridge University Press , 2007.

[36] D. Masse, „Bluetooth™: Connect Without Cables," *Microwave Journal ,* Bd. Vol.44(10), p. 208, 2001.

[37] N. J. Muller, Bluetooth, Bonn: MITP-Verl., 2001.

[38] M. Werner, Signale und Systme, Wiesbaden: Vieweg+ Teubner | GWV Fachverlage GmbH, 2008.

[39] J. Teich und H. Christian, Digitale Hardware/Software-Systeme, Berlin Heidelberg: Springer-Verlag, 2007.

[40] P. Berg und S. Michael, „Dipol modes of eye movements and blinks," *Electroencephalography and clinical Neurophysiology,* Nr. 79, pp. 36-44, 1991.

[41] A. Prechtl, *Vorlesungen über die Grundlagen der Elektrotechnik Band1,* Wien: Springer, 1994.

[42] Texas Instruments, „ADS129x Low-Power, 8-Channel, 24-Bit Analog Front-End for Biopotential Measurements," 01 2010. [Online]. Available: https://www.ti.com/lit/gpn/ads1298. [Zugriff am 10 06 2015].

[43] P. A. Höher, Grundlagen der digitalen Informationsübertragung, Wiesbaden: Vieweg+Teubner Verlag / Springer Fachmedien Wiesbaden GmbH, Wiesbaden , 2011.

[44] Google LCC, „Google Source for Android," The Android Open Source Project, 2009. [Online]. Available: https://android.googlesource.com/platform/development/+/eclair-passion-release/samples/BluetoothChat/src/com/example/android/BluetoothChat. [Zugriff am 01 04 2020].

[45] ISO, *ISO/IEC 11578:1996 Information technology - Open System Interconnection - Remote Procedure Call (RPC),* Genf, Schweiz: ISO , 1996-12.

[46] N. Fellows, „androidplot.com," 2015. [Online]. Available: http://androidplot.com/. [Zugriff am 05 03 2020].

[47] U. K. Che, C. K. Lao, S. H. Pun, P. U. Mak, F. Wan und M. I. Vai, „Portable Heart Rate Detector Based on Photoplethysmography with Android Programmable Devices," in *2012*

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

35th International Conference on Telecommunications and Signal Processing (TSP), Prague, 2012.

[48] N. T. M. K. Lehmann T., „Implementation of Dynamic Plotting for a Ventilation Simulator on Android mobile Devices," *Biomedizinische Technik,* p. 57 Suppl 1, 6 09 1012.

[49] MathWorks, „MathWorks," 5 10 2018. [Online]. Available: https://de.mathworks.com/help/matlab/.

[50] M. Werner, Signale und Systeme Lehr- und Arbeitsbuch mit MATLAB®-Übungen und Lösungen, Wiesbaden: GWV Fachverlage GmbH, 2008.

[51] The MathWorks, Inc., „MathWorks," 03 2020. [Online]. Available: https://de.mathworks.com/help/pdf_doc/compiler/compiler.pdf. [Zugriff am 07 04 2020].

[52] ISO/IEC, 7498-1, ISO, 1994.

[53] P. Berg und M. Scherg, „Dipole modelling of eye activity and its application to the removal of eye artefacts from the EEG and MEG," *Clin. Phys. Physiol. Meas,* Nr. Vol. 12, pp. 49-54, 1991.

[54] D. Dr. Lingenhöhl, „Spektrumverlag," Spektrum der Wissenschaft Verlagsgesellschaft mbH, [Online]. Available: https://www.spektrum.de/lexikon/physik/dipolfeld/3129. [Zugriff am 10 03 2020].

[55] P. Trillenberg, „Elektrookulographie," *Das Neurophysiologie-Labor,* pp. 98-106, Nr. 34 2012.

[56] M. Meyer, Signalverarbeitung Analoge und digitale Signale, Systeme und Filter, Vieweg+Teubner Verlag, 2011.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.
The approved original version of this thesis is available in print at TU Wien Bibliothek.