



**HOCHSCHULE LANDSHUT**  
HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN

FAKULTÄT INFORMATIK

# **Bachelorarbeit**

Automobilinformatik

ENTWICKLUNG DER FIRMWARE UND  
VISUALISIERUNGSSOFTWARE EINES MOBILEN  
NAH-INFRAROT-SPEKTROMETERS ZUR  
MATERIALIDENTIFIKATION

Arne Kulinna

Matrikelnummer: 1086667

Abgabedatum: 12.04.2021

Betreuer: Prof. Dr. paed. Artem Ivanov

Zweitbetreuer: Prof. Dr. Markus Mock

# ERKLÄRUNG ZUR BACHELORARBEIT

Arne Kulinna

## Hochschule Landshut Fakultät Informatik

Hiermit erkläre ich, dass ich die Arbeit selbständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen oder Hilfsmittel benutzt, sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.

.....

(Datum)

.....

(Unterschrift des Studierenden)



# Abstract

*Für das zielgerichtete Recycling von Kunststoffen ist dessen Identifikation ein wichtiger Beitrag, um der voranschreitenden Umweltbelastung entgegenzuwirken. Allerdings sind die dazu notwendigen Verfahren meist kostenintensiven Geräten vorbehalten und somit für viele Anwendungsbereiche und Zielgruppen nur eingeschränkt verfügbar. Mit der stetigen Weiterentwicklung von modernen Technologien entstehen kontinuierlich neue Möglichkeiten in der Herstellung von Sensorik für die Materialidentifikation.*

*In dieser Arbeit wird mit einer modernen Sensortechnologie im Umfeld einfacher Komponentenstrukturen für die mobile Anwendung gearbeitet. Dabei ist das Ziel die Bereitstellung einer Softwarelösung für die Firmware eines Handgerätes, welches die Aufnahme spektraler Verteilungen von Kunststoffen und Textilien erlaubt, sowie die zugehörige Software zur Visualisierung der entsprechenden Messdaten.*

*Im Zuge der Softwareentwicklung werden dazu die Hintergründe des Messverfahrens sowie der Aufbau des zugrundeliegenden Messgerätes erarbeitet. Mit der durch die Software mögliche Visualisierung von Messergebnissen wird die Fähigkeit des Gesamtsystems betrachtet.*

*Die Resultate zeigen die Bereitstellung einer Systembasis für die weitere Optimierung sowie aufbauende Methoden zur automatisierten Materialidentifikation.*

# Vorwort

Diese Arbeit umfasst alle grundlegenden Elemente der Entwicklung des Gesamtsystems des mobilen Nah-Infrarot-Spektrometers und fasst daher parallel zu den wesentlichen Aspekten der Entwicklung der entsprechenden Software zusätzlich die notwendigen Informationen zur Weiterführung der Entwicklung des Systems auf. Da die Quelltexte der in dieser Arbeit entwickelten Software als vertraulich behandelt werden, erfolgt die Beschreibung der Software ausschließlich in generischer Form. Protokolle und Datenformate zur Adaption eigener Lösungen werden allerdings detailliert beschrieben und in den beigefügten Anlagen zusammengefasst.

Spezielle und nicht geläufige Ausdrücke werden bei der ersten Nennung in dieser Arbeit durch kursiven Schreibstil gekennzeichnet und im Glossar erläutert. Firmen sowie Produktnamen werden bei der ersten Nennung im Text durch Kapitälchen gekennzeichnet und im weiteren Verlauf in üblicher Gebrauchsform genannt. Die Identifikation von Hexadezimalzahlen erfolgt durch den angefügten Index „16“.

An dieser Stelle möchte ich mich für die enge Zusammenarbeit sowie fachliche und persönliche Unterstützung bei meinem Betreuer Prof. Dr. paed. Artem Ivanov bedanken. Weiterhin danke ich meinem Zweitbetreuer Prof. Dr. Markus Mock für die stetige Verfügbarkeit bei aufgetretenen Fragen und Herrn Gerhard Sattelberger als meinen Ansprechpartner für Arbeiten im Elektroniklabor an der Hochschule Landshut.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>VIII</b>
<b>Tabellenverzeichnis</b>	<b>XI</b>
<b>Abkürzungsverzeichnis</b>	<b>XII</b>
<b>Glossar</b>	<b>XIV</b>
<b>Symbolverzeichnis</b>	<b>XV</b>
<b>1 Einleitung</b>	<b>1</b>
<b>2 Grundlagen</b>	<b>3</b>
2.1 Theoretische Hintergründe . . . . .	3
2.1.1 Nah-Infrarot-Spektroskopie . . . . .	3
2.1.2 Micro-Electro-Mechanical-System . . . . .	6
2.1.3 Fabry-Pérot Interferometer . . . . .	7
2.2 Anwendungsspezifische Grundlagen . . . . .	8
2.2.1 NIR MEMS-FPI Spektrolsensor . . . . .	8
2.2.2 RAD Studio . . . . .	10
2.2.2.1 TeeChart-Bibliothek . . . . .	10
<b>3 Planung</b>	<b>11</b>
3.1 Analyse des mobilen Nah-Infrarot-Spektrometers . . . . .	12
3.1.1 DC/DC Step-Up Wandler . . . . .	16
3.1.2 Signalverstärker . . . . .	18
3.2 Auslegung des Messsystems . . . . .	20
3.2.1 Steuerspannung . . . . .	20
3.2.2 Probenbelichtung . . . . .	23
3.2.3 Temperatureinfluss . . . . .	24
3.2.4 Wellenlängenberechnung . . . . .	25
3.3 Messdatenkonvertierung . . . . .	27

---

<b>4</b>	<b>Design und Implementierung</b>	<b>29</b>
4.1	Firmware . . . . .	29
4.1.1	Programmstruktur . . . . .	30
4.1.2	Benutzerschnittstelle . . . . .	31
4.1.3	Batteriespannung . . . . .	33
4.1.4	Spektralsensor . . . . .	34
4.1.4.1	Wellenlängenfilter . . . . .	39
4.1.4.2	Sensorsignal . . . . .	40
4.1.4.3	Temperatursensor . . . . .	40
4.1.5	Probenbelichtung . . . . .	41
4.1.6	Datenübertragung . . . . .	42
4.1.6.1	Kommunikationsprotokoll . . . . .	43
4.1.6.2	Serielle Schnittstelle . . . . .	46
4.2	Visualisierungssoftware . . . . .	49
4.2.1	Programmstruktur . . . . .	49
4.2.2	Datenübertragung . . . . .	51
4.2.3	Datenverwaltung . . . . .	55
4.2.4	Benutzeroberfläche . . . . .	57
4.2.5	Visualisierung der Daten . . . . .	60
<b>5</b>	<b>Evaluation des Systems</b>	<b>65</b>
<b>6</b>	<b>Fazit</b>	<b>71</b>
	<b>Anhang</b>	<b>XVI</b>
A	Bilder des mobilen Nah-Infrarot-Spektrometers . . . . .	XVII
B	Schaltplan Leiterplatte . . . . .	XVIII
C	Sensor Final Inspection Sheet . . . . .	XIX
D	Sensor Reference Datasheet . . . . .	XX
E	Sensor Calibration Coefficients . . . . .	XXII
F	Erzeugung Steuerspannung . . . . .	XXIII
G	Halogen Lamp Selection . . . . .	XXIV
H	Temperature Calculations . . . . .	XXV
I	Message Payload Content . . . . .	XXVII

---

J	Quick-Start User Guide . . . . .	XXIX
K	Klassen der Visualisierungssoftware . . . . .	XXX
L	File Content Structure . . . . .	XXXI
<b>Literaturverzeichnis</b>		<b>XXXII</b>



# Abbildungsverzeichnis

2.1	Spektrum elektromagnetischer Wellen. Der für den Menschen sichtbare Spektralbereich ist nur ein sehr geringer Bereich in diesem. In Anlehnung an [28]. . . . .	3
2.2	NIR-Spektrum verschiedener Polymere. Markant sind ausgeprägte Absorptionseigenschaften am Beispiel von PVC, Acryl, PET und PS [19]. . . . .	5
2.3	Schematische Funktionsweise eines FPI. Die beiden teildurchlässigen Spiegel $S$ sind planparallel mit einem Abstand $d$ angeordnet. In Anlehnung an [29]. . . . .	7
2.4	Analysierbare Medien der jeweiligen Hamamatsu-Sensoren welche den Wellenlängenbereichen farblich zugeordnet sind. Anlehnung an [2]. . . . .	8
2.5	Schematischer Aufbau eines Hamamatsu NIR MEMS-FPI Spektralsensors [19]. . . . .	9
3.1	In der Hand gehaltenes mobiles NIR-Spektrometer mit sichtbarer Unterseite und dessen Sensorik. . . . .	12
3.2	Elektronik des mobilen NIR-Spektrometers. Oberseite der im Gehäuse befindlichen Leiterplatte mit Mikrocontroller, Benutzerschnittstelle und Bluetooth-Modul (A). Unterseite mit Sensorik und Batterie der im Gehäuse befindlichen Leiterplatte (B). . . . .	13
3.3	Schematische Darstellung der auf der Leiterplatte vorhandenen Komponenten und Module sowie deren elektrische Verbindungen. Pfeile zeigen die Richtung des Informationsflusses an. . . . .	14
3.4	Qualitativer Verlauf der Steuerspannung des Sensors. Dargestellt ist ein schematischer Ausschnitt von zwei Spannungsänderungen. . . . .	21
3.5	Qualitative Verläufe der Steuerspannung. Diese verläuft zwischen maximaler Steuerspannung (Rot) und minimaler Steuerspannung (Grün). . . . .	22

---

3.6	Idealisierter Temperaturverlauf bei eingeschalteten Halogenlampen, gemessen durch den im Sensor integrierten NTC. Verwendet wurden vier Lampen des Typs 7361 bei $U_{Lampen} = 3,14\text{ V}$ . . . . .	25
4.1	Zustandsdiagramm des hierarchischen Zustandsautomaten der Systemzustände. Die Zustandsbezeichnungen (blau) entsprechen den im Quelltext verwendeten. . . . .	30
4.2	Qualitativer Verlauf des Signals eines Taster bei Betätigung. Der stabile Signalzustand wird nach Ablauf der Totzeit abgefragt. .	31
4.3	Daten eines Sample Points. Die eingelesene Steuerspannung des FPI (FPV), das Signal der Photodiode (Signal) und der Temperaturwert (Temp) benötigen 5 Byte Speicher pro Sample Point. . . . .	35
4.4	Schematische Darstellung der Synchronisation des SD-ADC mit dem Programmablauf. . . . .	38
4.5	Aufbau eines Frames. SOF und EOF sind als konstante Hexadezimalwerte angegeben. Die Byte-Reihenfolge ist entsprechend dem Mikrocontroller <i>Little-endian</i> . . . . .	44
4.6	Qualitativer Ablauf für das Senden eines Frames. Die farblichen Markierungen der einzelnen Daten entsprechen den in Abbildung 4.5 verwendeten. In den Prozess involviert sind die DMA-Peripherie, UART-Peripherie und CRC-Peripherie während TX den eigentlichen Datenbus kennzeichnet. . . . .	47
4.7	Zustandsdiagramm des Automaten zum Parsen von empfangenen Daten der UART-Schnittstelle. Orange Beschriftungen entsprechen Übergangsbedingungen. Kursive Beschriftungen entsprechen Aktionen. . . . .	48
4.8	Klassendiagramm der Visualisierungssoftware mit abstrahierten Klasseninhalten. . . . .	50
4.9	Aktivitätsdiagramm der Funktionen OpenDevice und SendData aus der Klasse TConnectionManager. . . . .	52

---

4.10	Qualitative Übersicht des zeitlichen Ablaufs der Datenübertragung einer Referenzmessung. Dargestellt sind die ausgehend von der Visualisierungssoftware gesendeten und empfangenen Nachrichten sowie der Zeitpunkt der Erzeugung der Instanz von TReference (violett). . . . .	54
4.11	Designkonzept der Benutzeroberfläche. Ansicht zum Verbinden mit einem Spektrometer (Links). Visualisierung und Steuerung von Messungen im Hauptbildschirm (Mitte). Geöffnetes Menü für weitere Funktionen (Rechts). . . . .	58
4.12	Darstellung der aufgenommenen spektralen Verteilungen von Baumwolle (blau) und Polyester (orange) in der Vergleichsansicht der Visualisierungssoftware auf einem Endgerät mit Windows-Betriebssystem. . . . .	63
5.1	Spektrale Verteilung von Baumwolle, Polyester und Wolle [19].	67
5.2	Auszug aus der Visualisierungssoftware mit Darstellung der spektralen Verteilungen von Baumwolle und Polyester. . . . .	67
5.3	Rohdaten von 15 Messungen von PP mit einem Messintervall von einer Minute. . . . .	69
5.4	Rohdaten der Messungen einer undefinierten Probe mit und ohne Erdung des Spektrometers. . . . .	70

# Tabellenverzeichnis

4.1	RGB-LED Signale. . . . .	32
4.2	Frame Type Codes. Die Richtung ist entsprechend des mobilen NIR-Spektrometers zu betrachten. . . . .	45
4.3	Bedeutung der Bluetooth-Symbole in der Navigationsleiste. .	59

# Abkürzungsverzeichnis

°C	Grad Celsius.
ADC	Analog-Digital-Converter.
ASCII	American Standard Code for Information Interchange.
CPU	Central Processing Unit.
CRC	Cyclic Redundancy Check.
CSV	Comma-Separated Value.
DAC	Digital-Analog-Converter.
DMA	Direct Memory Access.
EOF	End of Frame.
FPI	Fabry-Pérot Interferometer.
FPU	Floating-Point Unit.
GPIO	General-Purpose Input / Output.
GUI	Graphical User Interface.
IDE	Integrated Development Environment.
IEEE	Institute of Electrical and Electronics Engineers.
InGaAs	Indiumgalliumarsenid.
IR	Infrarot.
ISR	Interrupt Service Routine.

---

LED	Leuchtdiode.
LUT	Lookup Table.
MEMS	Micro-Electro-Mechanical-Systems.
NIR	Nah-Infrarot.
NIRS	Nah-Infrarot-Spektroskopie.
NTC	Negative Temperature Coefficient Thermistor.
OPV	Operationsverstärker.
PET	Polyethylenterephthalat.
PP	Polypropylen.
PS	Polystyrol.
PVC	Polyvinylchlorid.
PWM	Pulsweitenmodulation.
RGB	Rot-Grün-Blau.
RTL	Run-Time Library.
SD	Sigma-Delta.
SNV	Standard Normal Variates.
SOF	Start of Frame.
SPP	Serial Port Profile.
SRAM	Static Random-Access Memory.
UART	Universal Asynchronous Receiver Transmitter.
UI	User Interface.
USB	Universal Serial Bus.

# Glossar

Baudrate	Anzahl an Symboländerungen pro Zeiteinheit auf einem Datenbus.
Chemometrie	Das Lösen chemischer Problemstellungen durch Anwendung mathematischer und statistischer Methoden. Häufig findet die Chemometrie Anwendung im Bereich der Datenanalyse chemischer Experimente [16, S. 302].
Cloud	Netzwerk von Servern die dem Anwender Dienste über das Internet zur Verfügung stellen.
Firmware	Software die vom Hersteller fest in einem System eingebettet ist.
Hamming-Distanz	Anzahl an Bits die zwei binäre Zeichenketten voneinander unterscheiden [14].
Little-endian	Byte-Reihenfolge von im Speicher abgelegten Daten bei der die kleinste Speicheradresse einer Variable das niederwertigste Byte und die größte Speicheradresse entsprechend das höchstwertigste Byte enthält.
Parser	Computerprogramm zum Separieren und Umwandeln von Daten in ein zur Weiterverarbeitung geeignetes Format.
Prellen	Durch mechanische Eigenschaften eines elektromechanischen Kontaktes kurzzeitig nach Betätigung zufällig hervorgerufene Schaltwechsel.

# Symbolverzeichnis

Symbol	Beschreibung	Einheit
$\Delta\lambda_{FWHM}$	Als maximal angenommene spektrale Auflösung des Sensors	<i>nm</i>
$A$	Extinktion	1
$I$	Intensität der detektierten Strahlung	$W \cdot m^{-2}$
$I_0$	Intensität der einwirkenden Strahlung	$W \cdot m^{-2}$
$U_{\lambda 1550nm}$	Steuerspannung des FPI bei 1550nm	<i>V</i>
$U_{\lambda 1850nm}$	Steuerspannung des FPI bei 1850nm	<i>V</i>
$U_{AVDD}$	Analoge Betriebsspannung	<i>V</i>
$U_{Dunkel}$	Durch den Dunkelstrom der Photodiode bedingte Spannung	<i>V</i>
$U_{FPmax}$	Maximal erzeugbare Steuerspannung	<i>V</i>
$U_{FPmin}$	Minimal erzeugbare Steuerspannung	<i>V</i>
$U_{Licht}$	Durch das einfallende Licht auf die Photodiode bedingte Spannung	<i>V</i>
$U_{SD}$	Spannung am Eingang des SD-ADC	<i>V</i>



# 1 Einleitung

Moderne Sensortechnologien mit kleinem Formfaktor erlauben die Applikation von auf der Nah-Infrarot-Spektroskopie (NIRS) basierenden mobilen Messsystemen und somit unter anderem die Identifikation von Kunststoffen und Textilien. Während der Markt an solchen Messsystemen zur Zeit wenige kostenoptimierte Lösungen bietet, bleiben diese einem kommerziellen Umfeld vorbehalten.

Im Jahr 2014 hat das Unternehmen CONSUMER PHYSICS INC. die Entwicklung eines mobilen „Mikro-Spektrometers“ bekanntgegeben [15]. Dabei wurde mit der Analyse von Inhaltsstoffen und Zusammensetzungen von Objekten aller Bereiche des privaten Lebens geworben und sich somit von industriellen und komplexen Lösungen distanziert. Für die Analyse eines Objektes waren entsprechende Referenzdaten und dessen Bereitstellung über eine Cloud notwendig. Die Ansprüche der Kunden an das Produkt und die Menge der verfügbaren Daten konnten letztendlich nicht befriedigt werden, weshalb sich Consumer Physics daraufhin auf die Analyse von Agrarprodukten spezialisiert hat.

Hingegen fokussiert sich die aktuelle mobile Lösung der TRINAMIX GMBH auf kommerzielle Kunden und deren spezifische Anwendungen [4]. Bisher publizierte Anwendungen umfassen die Identifikation von Kunststoffen sowie die Analyse von Kaffeebohnen, welche ebenfalls die aktive Verbindung mit einer Cloud erfordern. Die Produkte beider Unternehmen basieren auf der Nah-Infrarot (NIR)-Spektroskopie, wobei nähere Informationen zu den verwendeten Sensoren oder dem Funktionsprinzip nicht öffentlich verfügbar sind.

Diese Arbeit ist Teil der Entwicklung eines einfachen und kostenoptimierten mobilen NIR-Spektrometers, welches auf Basis einer modernen Sensorbaureihe von HAMAMATSU PHOTONICS K.K. die Identifikation von Kunststoffen und Textilien anstrebt. Dabei bildet die Entwicklung der entsprechenden Software zum Betrieb des Spektrometers und der Visualisierung der Messdaten die Basis zur Bereitstellung des Gesamtsystems, dessen Funktionsfähigkeit dadurch in zukünftigen Arbeiten ausgebaut und optimiert werden kann. Priorität ist die drahtlose Kommunikation der zwei Softwareapplikationen, die aus der *Firmware* des Spektrometers sowie

der Visualisierungssoftware auf einem unabhängigen Endgerät mit BLUETOOTH-Schnittstelle bestehen. Durch die Begrenzung des Umfangs dieser Arbeit auf die Visualisierung spektraler Verteilungen ohne deren weitergehende analytische Auswertung besteht zudem keine Notwendigkeit einer Referenzdatenbank (Cloud).

Während Hamamatsu zum Zeitpunkt des Erarbeitungsbeginns dieser Arbeit Informationen zu einem proprietären NIR-Spektrometer unter Verwendung der genannten Sensorbaureihe als vertraulich behandelt hat, wurde das entsprechende Gerät im Verlauf der Erarbeitung veröffentlicht [3]. Obwohl der Funktionsumfang mit den Zielen dieser Arbeit vergleichbar ist, verwendet die proprietäre Lösung von Hamamatsu eine kabelgebundene Verbindung zu einem Computer mit eingeschränkter Betriebssystemkompatibilität. Die aus der Veröffentlichung resultierende Aktualisierung von technischen Unterlagen des Sensors erlaubte den Zugriff auf weitere Informationen über dessen Applikation.

Die Inhalte dieser Arbeit beginnen mit der Beschreibung von Grundlagen, die für das Verständnis der behandelten Thematiken notwendig sind, in Kapitel 2. In der anschließenden Planung in Kapitel 3 werden die zugrundeliegende Hardware analysiert und Auslegungseigenschaften des Gesamtsystems erarbeitet. Daraufhin folgt die Beschreibung des Designs und der Implementierung von sowohl Firmware als auch Visualisierungssoftware in Kapitel 4. Basierend auf der Evaluation in Kapitel 5 wird abschließend die Fähigkeit des Gesamtsystems durch Anwendung der entwickelten Software bestimmt.

## 2 Grundlagen

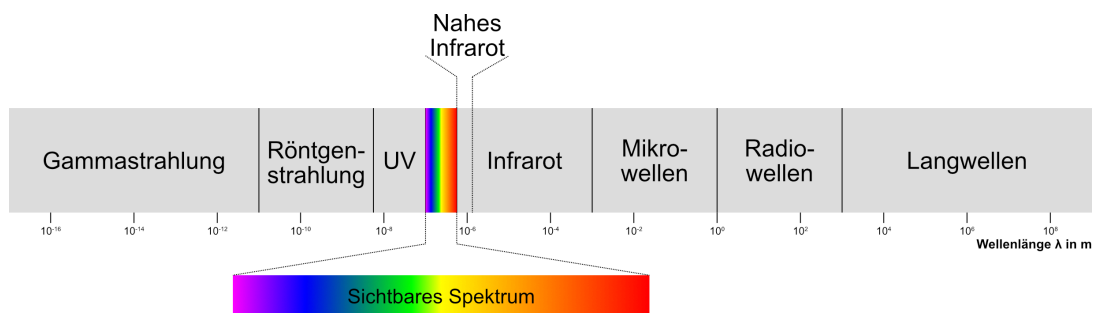
Ein wesentlicher Bestandteil für das Verständnis der Erarbeitung und des Ziels dieser Arbeit sind theoretische und anwendungsspezifische Grundlagen zu relevanten Methoden und Systemen.

### 2.1 Theoretische Hintergründe

Theoretische Grundlagen bilden die Basis für die thematische Einordnung der Umfänge sowie einen Überblick über technische Prinzipien dieser Arbeit. Solche werden in diesem Abschnitt erläutert.

#### 2.1.1 Nah-Infrarot-Spektroskopie

Die NIRS ist ein Verfahren auf Basis der optischen Spektroskopie, welche Prozesse zur qualitativen und quantitativen Analyse von Materie beschreibt [25]. Dabei wird die Wechselwirkung zwischen elektromagnetischer Strahlung und Materie verwendet, um Informationen über einen Stoff zu erhalten. Speziell bei der NIRS wird die elektromagnetische Strahlung in Form von Licht im Spektralbereich des NIR zwischen 800 nm und 2500 nm genutzt, welches sich außerhalb des niederfrequenten Randes des für den Menschen sichtbaren Spektrums befindet. Abbildung 2.1 zeigt einen Überblick über das Spektrum elektromagnetischer Wellen, deren Bezeichnungen und zugehörige Wellenlängen.



**Abb. 2.1:** Spektrum elektromagnetischer Wellen. Der für den Menschen sichtbare Spektralbereich ist nur ein sehr geringer Bereich in diesem. In Anlehnung an [28].

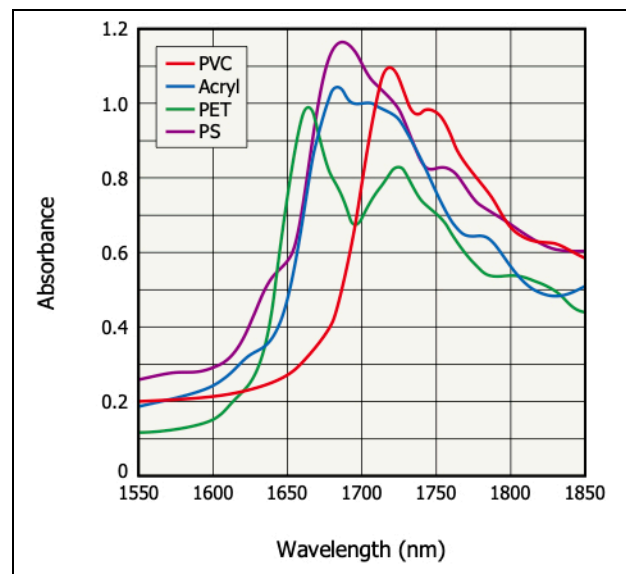
Als Vorteil der optischen Spektroskopie stellt sich sowohl die geringe Anforderung an die Probe, als auch die schnelle Analyse heraus. Diese erfolgt im NIR weder destruktiv noch invasiv und hat somit keinen verändernden Einfluss auf das Testobjekt (vgl. [25], Kapitel 9). Aufgrund dessen wird die NIRS in einer Vielzahl verschiedener Branchen angewendet. Dazu zählen unter anderem die Forschung, Medizin, Landwirtschaft und Industrie. Neben der quantitativen Analyse von organischen Proben, durch die sich beispielsweise der Wasser-, Protein- oder Fettgehalt landwirtschaftlicher Produkte bestimmen lässt, ist zudem die qualitative Analyse, wie die Identifikation verschiedener Polymere, möglich.

Physikalisch basiert das Prinzip der NIRS auf der Anregung von Molekülschwingungen durch elektromagnetische Bestrahlung. Dabei kommt es zu Absorption und Emission von Photonen, deren Energie beziehungsweise Wellenlänge für den analysierten Stoff charakteristisch ist. Die durch spezielle Messeinrichtungen detektierte Strahlung resultiert in einer spezifischen spektralen Verteilung. Für die Aufnahme einer solchen Verteilung bestehen verschiedene Varianten. Bei der Transmissionspektroskopie wird eine Probe durchleuchtet und das transmittierte Licht detektiert. Dafür muss die Probe Anforderungen hinsichtlich ihrer Transparenz erfüllen, da undurchsichtige Proben das gesamte Licht absorbieren. Im Gegensatz dazu erfordert die Messung bei der Reflexionspektroskopie keine durchsichtige Probe. In Reflexion erfolgt eine Messung durch Beleuchtung der Probe und Detektion der reflektierten Strahlung.

Während sich über optische Vergleiche der spektralen Verteilung erste qualitative Schlüsse über eine Probe treffen lassen, sind für quantitative Analysen Methoden der *Chemometrie* erforderlich [28, S. 6]. Begründet ist dies durch wesentlich mehr als die hier genannten Effekte, die in der Praxis Einfluss auf die Wechselwirkung der elektromagnetischen Strahlung mit der Materie haben. In dieser Arbeit werden die Methoden zur quantitativen Analyse sowie weitere physikalische Effekte nicht weiter berücksichtigt.

Dargestellt wird die spektrale Verteilung in einem Graphen, bei dem die optische Dichte beziehungsweise Extinktion (engl.: absorbance) gegenüber der Wellenlänge (engl. wavelength) aufgetragen ist [25, S. 334]. Obwohl sich die Resultate einer Messung in Transmission von einer Messung in Reflexion stark unterscheiden und

diese keinen direkten Vergleich zulassen, wird üblicherweise für beide Ergebnisse die Darstellung in Extinktion verwendet (vgl. [28] Abschnitt 1.4).



**Abb. 2.2:** NIR-Spektrum verschiedener Polymere. Markant sind ausgeprägte Absorptionseigenschaften am Beispiel von Polyvinylchlorid (PVC), Acryl, Polyethylenterephthalat (PET) und Polystyrol (PS) [19].

Unter Extinktion ist ein Maß für die Abschwächung einer Strahlung zu verstehen, nachdem diese ein Medium durchquert hat (vgl. [25] Abschnitt 4.1.2). Dazu zählt die gesamte Lichtschwächung, die durch Streuung, Reflexion und Absorption entsteht. Die Berechnung der Extinktion basiert auf der Transmission  $T$ , wobei diese das Verhältnis der Intensitäten von der am Detektor einfallenden Strahlung  $I$  zu der auf die Probe einwirkenden Strahlung  $I_0$  beschreibt:

$$T = \frac{I}{I_0} \quad (2.1)$$

Für die Extinktion  $A$  gilt:

$$A = -\log_{10} T = \log_{10} \frac{I_0}{I} \quad (2.2)$$

Abbildung 2.2 zeigt die spektrale Verteilung mehrerer Polymere, die durch eine Messung in Transmission ermittelt wurden. Dabei ist wie zuvor beschrieben die Extinktion gegenüber der Wellenlänge aufgetragen. Eindeutig zu erkennen sind die ausgeprägten Absorptionseigenschaften der verschiedenen Kunststoffe im Bereich zwischen 1650 nm und 1750 nm, welche einen für den Polymer charakteris-

tischen Verlauf aufzeigen. Im Gegensatz zu den guten Analyseeigenschaften bei organischen Proben und Polymeren bietet die NIRS schlechte Eigenschaften für die Analyse von Gasen. „Der wichtigste Spektralbereich, der für gasanalytische Zwecke genutzt wird, liegt zwischen 3  $\mu\text{m}$  und 12  $\mu\text{m}$  ...“ [29, S. 336]. Nahezu alle Gase haben in diesem Wellenlängenbereich eine ausgeprägte Absorptionsbande. Wie in Abbildung 2.1 gezeigt befindet sich diese zwar im Spektralbereich des Infrarot (IR), allerdings außerhalb des NIR.

### 2.1.2 Micro-Electro-Mechanical-System

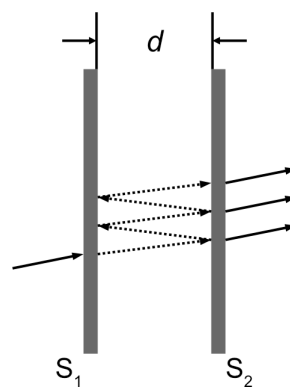
Der Begriff Micro-Electro-Mechanical-Systems (MEMS) bezieht sich auf ein Fertigungsverfahren, welches die Herstellung von elektromechanischen Systemen in ähnlicher Form wie von integrierten Schaltkreisen erlaubt [24, S. 2]. Dabei ist es möglich, neben komplexen mechanischen Systemen zusätzlich die zur Steuerung oder Auswertung notwendige Elektronik in der Größenordnung von Mikrometern in einem System zu kombinieren. Aufgrund der ähnlichen Herstellungsverfahren sind die Produktionsprozesse leicht miteinander zu verknüpfen und kostengünstig zu realisieren. Obwohl sich der Begriff MEMS auf das Fertigungsverfahren bezieht, werden auch die dadurch hergestellten Systeme als MEMS bezeichnet.

Anwendung finden MEMS besonders im Bereich der Sensorik und Aktuatorik, wie zum Beispiel bei Beschleunigungssensoren (vgl. [17] Abschnitt 1.1). Dabei wird durch die Kombination von einer mechanisch fixierten Platte und einer mechanisch beweglichen Platte ein Kondensator gebildet. Bei der Beschleunigung des Sensors und der daraus resultierenden Einwirkung der Beschleunigung auf die Platten verändert sich die Distanz zwischen diesen und somit die Kapazität des Kondensators. Die in einem als MEMS gefertigten Beschleunigungssensor integrierte Elektronik wertet diese aus und überträgt die relevanten Informationen über eine geeignete Schnittstelle.

Ein weiteres Beispiel für die Nutzung von MEMS in der Aktuatorik sind die Düsen eines Tintenstrahldruckers [18]. Aufgrund der Fertigungsmöglichkeit von Düsen in der Größenordnung von Mikrometern und der dichten Platzierung dieser wird eine hohe Druckqualität bei gleichzeitig hohem Kontrast erreicht.

### 2.1.3 Fabry-Pérot Interferometer

Das Fabry-Pérot Interferometer (FPI) ist ein Vielstrahlinterferometer, welches aufgrund des einfachen Aufbaus und der hohen Kontraststärke der Interferenzfiguren häufig als Interferenzfilter verwendet wird (vgl. [25] Abschnitt 3.6). Technisch basiert es in der einfachsten Form auf zwei teildurchlässigen, parallel zueinander angeordneten Spiegeln. Einfallendes Licht wird, wie in Abbildung 2.3 dargestellt, zum großen Teil zwischen den Spiegeln reflektiert.



**Abb. 2.3:** Schematische Funktionsweise eines FPI. Die beiden teildurchlässigen Spiegel  $S$  sind planparallel mit einem Abstand  $d$  angeordnet. In Anlehnung an [29].

Jene Lichtstrahlen, welche die Resonanzbedingung des Filters nicht erfüllen, werden durch destruktive Interferenz nahezu vollständig ausgelöscht. Durch eine mechanische Änderung der Distanz zwischen den Spiegeln lässt sich das Filter dynamisch auf eine gewünschte Filter-Wellenlänge einstellen und ist idealerweise nur für diese durchlässig. Real wird ein schmales Band an Wellenlängen gefiltert. Für diese Arbeit ist die Bezeichnung einer gefilterten Wellenlänge allerdings immer als idealer Transmissions-Spitzenwert anzunehmen.

Sogenannte durchstimmbare FPI mit variablem Spiegelabstand lassen sich als MEMS fertigen, wodurch sie für die Anwendung in Sensoren mit kleiner Bauform geeignet sind [18]. Die Variation des Spiegelabstandes ist dabei durch elektrische Spannung vorzunehmen.

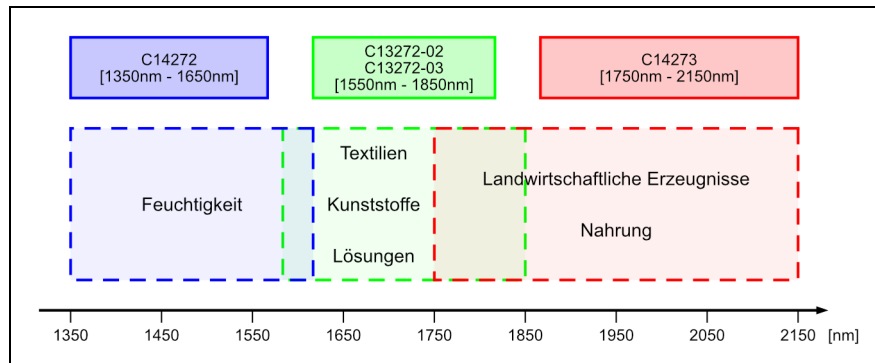
## 2.2 Anwendungsspezifische Grundlagen

Anwendungsspezifische Grundlagen bieten einen Überblick über die in dieser Arbeit verwendeten Hardware-Komponenten, Entwicklungswerkzeuge und Software-Bibliotheken. Im Folgenden werden diese beschrieben näher.

### 2.2.1 NIR MEMS-FPI Spektroskop

Das Unternehmen Hamamatsu bietet Sensoren für die spektrale Analyse im NIR an [18]. Speziell die sogenannten MEMS-FPI Spektroskop unterscheiden sich durch die kompakte Bauform und die sequentielle Vermessung des Wellenlängenbereiches von anderen Spektroskop.

Seit der Markteinführung im Jahr 2015 ist Hamamatsu neben dem deutschen Unternehmen INFRA TEC GMBH einer der wenigen Hersteller von Sensoren dieser Art [18]. Für jeden Sensor gibt Hamamatsu einen geeigneten Einsatzbereich an, welcher vor allem aufgrund des spektralen Bereiches zuzuordnen ist.

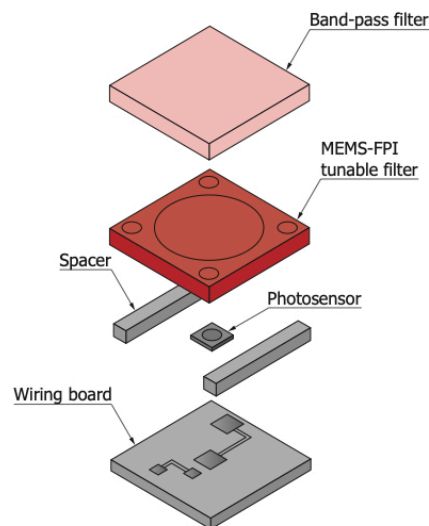


**Abb. 2.4:** Analysierbare Medien der jeweiligen Hamamatsu-Sensoren welche den Wellenlängenbereichen farblich zugeordnet sind. Anlehnung an [2].

Wie in Abbildung 2.4 dargestellt ist, decken die vier verfügbaren Sensoren zusammen einen Wellenlängenbereich von 1350 nm bis 2150 nm ab und teilen diesen in drei Einsatzbereiche ein. Somit ist neben der Feuchtigkeitsmessung und der Analyse von Textilien, Kunststoffen und Lösungen auch die Inhaltsanalyse von landwirtschaftlichen Erzeugnissen oder Nahrung durch Einsatz des jeweils diesem Zweck zugeordneten Sensors möglich.



Die Betrachtung des schematischen Aufbaus in Abbildung 2.5 zeigt die Kombination des im Abschnitt 2.1.3 beschriebenen FPI als MEMS gefertigtes Bauteil mit einer Indiumgalliumarsenid (InGaAs)-Photodiode als wesentlichen Bestandteil des Sensors. Für die Darstellung ist anzunehmen, dass einfallendes Licht von oben auf das Bandpass-Filter einfällt, welches den spektralen Bereich des Lichts auf den sensitiven Wellenlängenbereich des Sensors beschränkt. Durch eine Variation des Spiegelabstandes des als Interferenzfilter wirkenden FPI ist dieses, wie bereits in Abschnitt 2.1.3 beschrieben, idealerweise nur für eine Wellenlänge durchlässig. Somit lässt sich die Intensität des auf die im NIR-Spektralbereich sehr empfindliche InGaAs-Photodiode einfallenden Lichts zu einer gefilterten Wellenlänge des FPI zuordnen. Dementsprechend ist ein spektraler Bereich durch schrittweise Detektion der Lichtintensität einzelner Wellenlängen zu erfassen.



**Abb. 2.5:** Schematischer Aufbau eines Hamamatsu NIR MEMS-FPI Spektralsensors [19].

Die aufgrund der sequentiellen Vermessung eines spektralen Bereiches geringe Anzahl an notwendigen Bauteilen resultieren in einem einfachen Aufbau des Sensors. In Kombination mit der Herstellung als MEMS ergeben sich dadurch geringe Produktionskosten für die Sensoren, welche aufgrund der kleinen Baugröße besonders vorteilhaft in mobilen Anwendungen sind [18].

## 2.2.2 RAD Studio

RAD STUDIO ist ein von EMBARCADERO TECHNOLOGIES, INC. entwickeltes Integrated Development Environment (IDE), das es erlaubt, native Applikationen für Plattformen mit MICROSOFT WINDOWS, GOOGLE ANDROID, APPLE MACOS, APPLE IOS und LINUX als Betriebssystem zu entwickeln [6]. Dabei können die verschiedenen nativen Anwendungen direkt aus einem Quellcode erzeugt werden. Während die sogenannte Delphi-Run-Time Library (RTL) sämtliche systemübergreifende Methoden bereitstellt, lässt sich das Graphical User Interface (GUI) einer Applikation durch das integrierte FIREMONKEY-Framework plattformübergreifend gestalten.

Unterstützt werden die objektorientierten Programmiersprachen OBJECT PASCAL beziehungsweise DELPHI und C++. Mit dem RAD Studio lässt sich der gesamte Entwicklungsprozess vom Entwurf des GUI über das Implementieren und Debuggen bis hin zum letztendlichen Bereitstellen der Anwendung abdecken.

Ein alternatives, weit verbreitetes Framework für plattformübergreifende Anwendungen ist QT [21]. Qt deckt mit seinen Entwicklungswerkzeugen gleichermaßen den gesamten Arbeitsumfang für die Anwendungsentwicklung ab. Im Gegensatz zum RAD Studio werden weitaus mehr Programmiersprachen und Plattformen unterstützt, darunter auch die Anwendung auf eingebetteten Systemen. So findet das Framework zum Beispiel Anwendung in Infotainment-Systemen von Großserien-Kraftfahrzeugen [5].

### 2.2.2.1 TeeChart-Bibliothek

Das Unternehmen STEEMA bietet mit der Softwarebibliothek TEECHART eine Erweiterung für verschiedene Frameworks und Programmiersprachen zur Darstellung von Diagrammen an [7]. Diese dient der einfachen graphischen Darstellung von Daten in dem GUI, wobei neben Ausführungen für JAVA, PHP und HTML5 auch das FireMonkey-Framework unterstützt wird. Eine in den Funktionen eingeschränkte Version der TeeChart-Bibliothek ist im RAD Studio integriert und bietet wesentliche Funktionen zum Anzeigen verschiedener Diagrammtypen.

## 3 Planung

Der Fokus dieser Arbeit liegt in der Entwicklung der Firmware und Visualisierungssoftware für das mobile NIR-Spektrometer, welches im anschließenden Abschnitt genauer beschrieben wird. Neben der Softwareentwicklung sind zudem die technischen Hintergründe und Verfahren zur Aufnahme der spektralen Verteilung einer Probe für die anschließende graphische Darstellung der Ergebnisse relevant. Diese sind zum einen essentiell für Entscheidungsfindungen in der Entwicklung, zum anderen dienen sie dem Verständnis der letztendlich ausgegebenen Informationen der Programme.

Bereits ohne genaue Kenntnis über das zugrundeliegende mobile NIR-Spektrometer sind Anforderungen an das Gesamtsystem zu differenzieren. Ein Anspruch an die Interaktion zwischen dem mobilen NIR-Spektrometer und der Visualisierungssoftware ist die drahtlose Kommunikation. Dabei ist allerdings keine autarke Anwendung des Spektrometers gefordert, weshalb die Nutzung lediglich bei bestehender Kommunikationsverbindung gewährleistet sein muss. Um eine Mobilität des Gesamtsystems zu erreichen, ist die Visualisierungssoftware plattformübergreifend für sowohl Windows-basierte Systeme, als auch für Android Smartphones zu entwickeln. Die in Abschnitt 2.2.2 beschriebene IDE in Verbindung mit Delphi und dessen Frameworks sind als Entwicklungswerkzeuge vorgegeben.

Entsprechend dient das durch die Firmware gesteuerte Spektrometer dem Sammeln von Informationen, während die Visualisierungssoftware dem Anwender auf einem Endgerät in Form eines Computers oder Smartphones die Darstellung und Verwaltung dieser ermöglicht. Das Bedienkonzept der Firmware und der Visualisierungssoftware ist kein relevanter Bestandteil der Planung, da der Fokus in diesem Kapitel auf den technischen Grundlagen der Hardware basiert und die detaillierte Form der Bedienung daher für die Implementierungen getrennt im Kapitel 4 beschrieben werden. Demnach werden nachfolgend die zugrundeliegende Hardware analysiert und anschließend alle technisch-relevanten Aspekte erläutert.

### 3.1 Analyse des mobilen Nah-Infrarot-Spektrometers

Das für diese Arbeit gegebene mobile NIR-Spektrometer besteht aus einer elektronischen Einheit, welche in einem Kunststoffgehäuse mit den Außenmaßen 120 mm x 65 mm x 40 mm (L x B x H) eingebracht ist. Abbildung 3.1 zeigt zum Größenvergleich das in der Hand gehaltene mobile NIR-Spektrometer. Weitere Bilder des Gerätes sind in Anlage A zusammengefasst. Auf der Unterseite des Gerätes befindet sich die Sensorik, welche als Kombination aus Sensor, Leuchtmitteln und den als Reflektor fungierenden Baugruppen zu verstehen ist. Der verwendete Hamamatsu NIR MEMS-FPI Spektralsensor C13272-03 entspricht der in Abschnitt 2.2.1 beschriebenen Baureihe und hat einen spezifizierten Wellenlängenbereich von 1550 nm bis 1850 nm bei einer typischen spektralen Auflösung zwischen 11 nm und 16 nm [20].

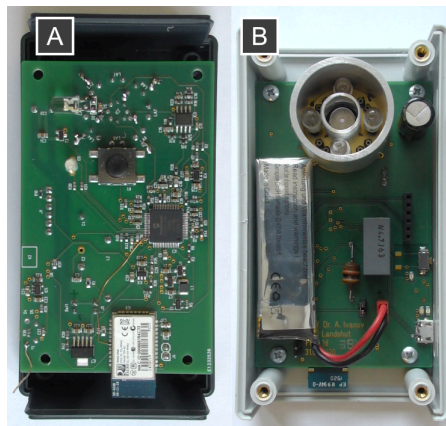


**Abb. 3.1:** In der Hand gehaltenes mobiles NIR-Spektrometer mit sichtbarer Unterseite und dessen Sensorik.

Innerhalb des Gehäuses des mobilen NIR-Spektrometers befindet sich die auf Abbildung 3.2 gezeigte Leiterplatte, welche von Prof. Dr. paed. Artem Ivanov entwickelt wurde. Auf dieser sind alle relevanten Schaltkreise, Komponenten und Module vorhanden, die zum Ansteuern und Einlesen des Hamamatsu NIR MEMS-FPI Spektralsensors, sowie der drahtlosen Bluetooth-Kommunikation mit einem Computer oder Smartphone notwendig sind.

In Kombination mit der im Gehäuse befindlichen wiederaufladbaren Batterie ist die mobile Anwendung ohne Kabelanbindung gewährleistet. Geladen wird die Batterie über die integrierte Universal Serial Bus (USB)-Schnittstelle, welche zudem die Kommunikation mit dem Mikrocontroller ermöglicht. Aufgrund der drahtlosen

Kommunikationsmöglichkeit zwischen dem Spektrometer und einem Computer oder Smartphone ist von der Nutzung der leitungsgebundenen Kommunikation über die USB-Schnittstelle in dieser Arbeit abzusehen. Anzumerken ist, dass bei entladener Batterie und gleichzeitiger Spannungsversorgung über die USB-Schnittstelle der Betrieb des Gerätes nur begrenzt möglich ist.



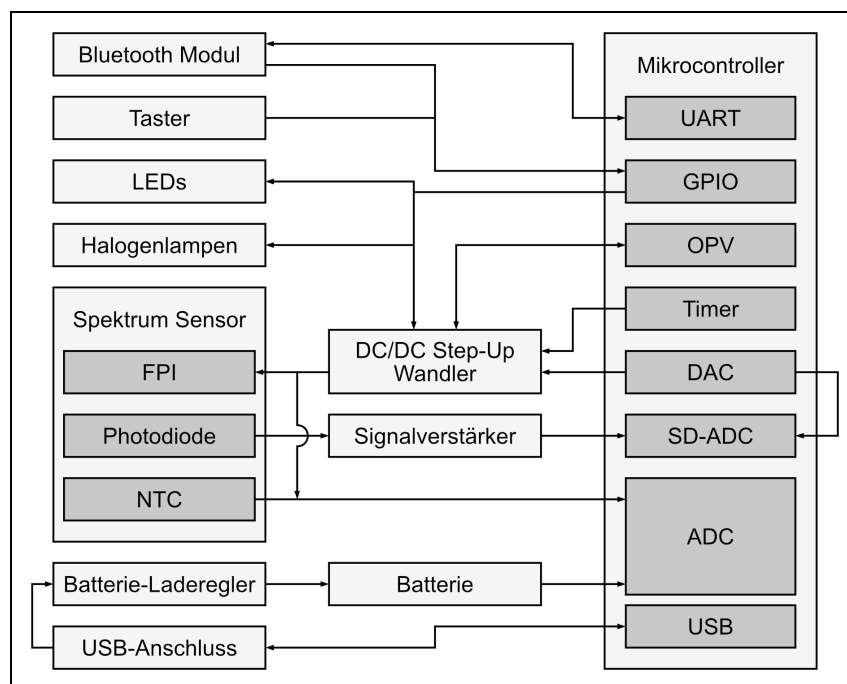
**Abb. 3.2:** Elektronik des mobilen NIR-Spektrometers. Oberseite der im Gehäuse befindlichen Leiterplatte mit Mikrocontroller, Benutzerschnittstelle und Bluetooth-Modul (A). Unterseite mit Sensorik und Batterie der im Gehäuse befindlichen Leiterplatte (B).

Abbildung 3.3 zeigt einen schematischen Überblick über die wesentlichen Schaltkreise beziehungsweise Module und deren elektrische Verbindungen, die auf der Leiterplatte vorhanden sind. Die zentrale Steuereinheit bildet ein PIC24FJ128GC006 Mikrocontroller des Halbleiterherstellers MICROCHIP TECHNOLOGY INC. mit 16 Bit-Architektur [12]. Dieser ist über eine Kontaktreihe auf der Leiterplatte mithilfe eines externen Programmieradapters und der Microchip MPLAB X IDE zu flashen und debuggen.

Entsprechend dem in der IDE installierten C-Compiler erfolgt die Implementierung der Firmware des Mikrocontrollers in der Programmiersprache C. Aufgrund der Betriebsspannung des Mikrocontrollers von 3,3 V ist für alle eingehenden und ausgehenden Signale des Mikrocontrollers maximal von einer Spannung von 3,3 V auszugehen, wobei gleichermaßen für die in Abschnitt 4 folgenden Berechnungen die analoge Betriebsspannung  $U_{AVDD}$  mit 3,3 V zu definieren ist.

Als hardwareseitige Benutzerschnittstelle sind zwei Taster sowie eine Rot-Grün-Blau (RGB)-Leuchtdiode (LED) auf der Leiterplatte vorhanden (vgl. Anlage B). Bei

beiden Tastern wird dem *Prellen* durch Kondensatoren entgegengewirkt. Taster K1 vereint das Einschalten der Spannungsversorgung sowie ein vom Mikrocontroller nutzbares Schaltsignal. Ein Betätigen von K1 bei ausgeschalteter Spannungsversorgung aktiviert die Spannungsversorgung aller Komponenten inklusive des Mikrocontrollers. Dieser muss einen General-Purpose Input / Output (GPIO)-Pin auf den „High“-Pegel setzen und somit die Selbsthaltung der Spannungsversorgung aktivieren. Ohne die Aktivierung der Selbsthaltung sind jegliche Komponenten, ausgenommen des Batterie-Ladereglers, nur im Falle der dauerhaften Betätigung von K1 mit Spannung versorgt. Dieses Verhalten ist sowohl beim Flashen als auch bei der Entwicklung der Firmware des Mikrocontrollers zu beachten. Aufgrund der Bauform und Lage des Tasters K2 ist dieser für das Zurücksetzen des Systems angedacht.



**Abb. 3.3:** Schematische Darstellung der auf der Leiterplatte vorhandenen Komponenten und Module sowie deren elektrische Verbindungen. Pfeile zeigen die Richtung des Informationsflusses an.

Zur drahtlosen Kommunikation zwischen mobilem Spektrometer und einem Endgerät mit Visualisierungssoftware ist das Bluetooth-Modul RN-42 von Microchip auf der Leiterplatte aufgebracht (vgl. Abbildung 3.1 B). Unter Verwendung von Bluetooth-Classic fungiert dieses in der Standard-Konfiguration als Brücke zwischen Universal Asynchronous Receiver Transmitter (UART)-Schnittstelle des Mi-

krocontrollers und einer emulierten seriellen Schnittstelle auf dem Endgerät [10]. Ermöglicht wird diese Funktion durch das Serial Port Profile (SPP), welches Protokolle und Prozeduren für die Emulation einer drahtgebundenen seriellen Schnittstelle über Bluetooth beschreibt [9].

Für die Nutzung des Bluetooth-Moduls RN-42 ist weder eine Konfiguration noch eine spezielle Form der Übertragungsdaten notwendig. Deshalb sind auch keine tiefgreifenden Kenntnisse von Bluetooth und dessen Protokolle für die Umsetzung der Kommunikation zwischen mobilem Spektrometer und Endgerät notwendig. Die Anbindung an den Mikrocontroller besteht aus den seriellen Kommunikationsleitungen sowie einer Signalleitung zur Feststellung des Verbindungsstatus (vgl. Anlage B). Letzteres Schaltsignal teilt dem Mikrocontroller mit, ob ein Endgerät verbunden und die Verbindung über das SPP hergestellt ist. Während bei der Entwicklung der Firmware lediglich von der Kommunikation über die UART-Schnittstelle auszugehen ist, ist der Zugriff auf die emulierte serielle Schnittstelle auf dem Endgerät von der verwendeten Programmiersprache und dessen Frameworks abhängig und wird daher in Abschnitt 4.2.2 weiter erläutert.

Eine Messung mit dem mobilen NIR-Spektrometer beschreibt das Aufnehmen des NIR-Spektrums einer Probe. Durch das Design der Leiterplatte ist bereits vorgegeben, dass eine Messung in Reflexion erfolgt. Die Vorteile dieses Verfahrens wurden bereits in Abschnitt 2.1.1 erläutert.

Auf Abbildung 3.2 ist der detaillierte Aufbau der auf der Leiterplatte verbauten Sensorik zu sehen. Die ringförmig um den Sensor angeordneten Lampen beleuchten die Probe, während das von dieser reflektierte Licht auf den zentral gelegenen Sensor einfällt. Sowohl die Lampen als auch den Sensor umgeben Aluminiumrohre, welche die direkte Einstrahlung des Lichts der Lampen auf den Sensor beschränken und gleichzeitig als Reflektor dienen. Genauere Erläuterungen zu den verwendeten Lampen folgen in Abschnitt 3.2.2. Für das Vermessen einer Probe ist die Sensorik des mobilen NIR-Spektrometers vollständig mit dieser zu bedecken. Ohne die vollständige Bedeckung der Sensorik mit der Probe wird das Messergebnis durch den Einfall von Fremdlicht auf den Sensor undefinierbar verändert.

Wesentlich für die Vermessung einer Probe ist das Ansteuern des Sensors sowie das Einlesen der gemessenen Signale. Wie bereits in Abbildung 3.3 ersichtlich, ist

der Sensor über einen DC/DC Step-Up Wandler und einen Signalverstärker an den Mikrocontroller angebunden. Die Eigenschaften und Funktionsweise dieser Schaltkreise sind essentiell für die Entwicklung der Firmware und werden daher in den folgenden Abschnitten spezifisch erläutert.

### 3.1.1 DC/DC Step-Up Wandler

Die zum Variieren des Spiegelabstandes des im Sensor befindlichen FPI benötigte Steuerspannung (vgl. Abschnitt 2.1.3) wird durch einen DC/DC Step-Up Wandler erzeugt. Dieser ist teilweise durch interne Peripherie des Mikrocontrollers aufgebaut und durch diese zu steuern (vgl. Abbildung 3.3). Neben einem Operationsverstärker (OPV) und einem Timer sind zusätzlich ein 10-bit Digital-Analog-Converter (DAC) und ein GPIO-Pin im Schaltkreis eingebunden, für die bereits eine optimale Konfiguration der Peripherie-Register vorliegt. Mittels der am DAC-Ausgang eingestellten Spannung ist die von dem DC/DC Step-Up Wandler zu erzeugende Spannung einzustellen, wobei das Verhältnis zwischen diesen Spannungen linear antiproportional ist. Im weiteren Verlauf ist als Steuerspannung stets die am Sensor angelegte Spannung zum Einstellen des Spiegelabstandes des FPI anzunehmen.

Dem Hamamatsu NIR MEMS-FPI Spektralsensor liegt ein sogenanntes „Final Inspection Sheet“ bei (siehe Anlage C). Darin vermerkt sind Referenzwerte des Sensors, welche unter Laborbedingungen bei 25 Grad Celsius ( $^{\circ}\text{C}$ ) Umgebungstemperatur ermittelt wurden. Elementar für die Ansteuerung des DC/DC Step-Up Wandlers ist die Größe der Steuerspannung des FPI bei minimaler Wellenlänge ( $U_{\lambda 1550\text{nm}}$ ) und maximaler Wellenlänge ( $U_{\lambda 1850\text{nm}}$ ). Da zwischen Steuerspannung und Wellenlänge ein nichtlineares antiproportionales Verhältnis besteht, ist  $U_{\lambda 1550\text{nm}}$  größer als  $U_{\lambda 1850\text{nm}}$  [20].

Aufgrund des sogenannten Pull-in Phänomens, bei welchem die beiden Spiegel des FPI bei einer zu hohen angelegten Steuerspannung nicht reversibel aneinander haften können, darf diese  $U_{\lambda 1550\text{nm}} + 0,5\text{ V}$  nicht überschreiten [19]. Zu beachten ist hier die Temperaturabhängigkeit des FPI, sodass bei Umgebungstemperaturen unter  $25\text{ }^{\circ}\text{C}$  auch  $U_{\lambda 1550\text{nm}}$  und somit die Schwelle zum Eintreten des Pull-in Phänomens sinkt. Deshalb ist die Auslegung der maximal durch den DC/DC Step-Up Wandler



erzeugbaren Spannung unbedingt unter Berücksichtigung von  $U_{\lambda 1550\text{nm}}$  und der Umgebungstemperatur zu treffen.

Um die Zerstörung des Sensors durch das Pull-In Phänomen zu vermeiden, ist die vom DC/DC Step-Up Wandler maximal erzeugte Spannung hardwareseitig durch eine Zener-Diode auf gemessene  $U_{FPmax} = 37,7\text{ V}$  begrenzt. Die minimal erzeugbare Spannung beträgt gemessene  $U_{FPmin} = 23,1\text{ V}$ . Der Vergleich mit dem im „Final Inspection Sheet“ angegebenen Werten für  $U_{\lambda 1550\text{nm}}$  und  $U_{\lambda 1850\text{nm}}$  zeigt, dass unter Berücksichtigung einer Umgebungstemperatur von  $25\text{ °C}$  daher nicht der gesamte Spektralbereich des Sensors ausgenutzt werden kann. Weitere Erläuterungen dazu folgen in Abschnitt 3.3.

Unter Betrachtung der spektralen Auflösung des gegebenen Sensors, welche bei einer gefilterten Wellenlänge von  $1625\text{ nm}$  etwa  $11\text{ nm}$  erreichen kann, ist für die weiteren Berechnungen  $\Delta\lambda_{FWHM} = 10\text{ nm}$  als maximale spektrale Auflösung anzunehmen [20]. Um über den gesamten Spektralbereich des Sensors dauerhaft die maximale spektrale Auflösung zu erhalten, muss die Steuerspannung des FPI mindestens auf  $n_{Sensor}$  Spannungswerte einstellbar sein:

$$n_{Sensor} = \frac{\lambda_{max} - \lambda_{min}}{\Delta\lambda_{FWHM}} = \frac{1850\text{ nm} - 1550\text{ nm}}{10\text{ nm}} = 30 \quad (3.1)$$

Auf Basis von  $n_{Sensor}$  ergibt sich die Spannungsdifferenz  $\Delta U_{Sensor}$  zwischen jeder Spannungs-Stufe:

$$\Delta U_{Sensor} = \frac{U_{\lambda 1850\text{nm}} - U_{\lambda 1550\text{nm}}}{n_{Sensor}} = \frac{38,079\text{ V} - 26,441\text{ V}}{30} = 0,388\text{ V} \quad (3.2)$$

Die maximale Anzahl an Spannungs-Stufen, auf welche sich der Ausgang des DC/DC Step-Up Wandlers einstellen lässt, ist durch den 10-bit DAC beschränkt:

$$n_{FP} = 2^{n_{bits}} = 2^{10} = 1024 \quad (3.3)$$

Unter Verwendung des Spannungsbereiches, der vom DC/DC Step-Up Wandlers tatsächlich erzeugt werden kann, berechnet sich die minimale Spannungsdifferenz zwischen zwei Stufen des Ausgangssignals zu:

$$\Delta U_{FPmin} = \frac{U_{FPmax} - U_{FPmin}}{n_{FP}} = \frac{37,7\text{ V} - 23,1\text{ V}}{1024} = 0,014\text{ V} \quad (3.4)$$

Daraus ergibt sich, dass die maximale spektrale Auflösung des Sensors ausgenutzt werden kann, da  $\Delta U_{FPmin} < \Delta U_{Sensor}$  erfüllt ist. Um bei einer Reduktion der Spannungsstufen die maximale spektrale Auflösung des Sensors stets auszunutzen ergibt sich für die minimale Anzahl der dafür notwendigen Spannungsstufen:

$$n_{FPmin} = \frac{U_{FPmax} - U_{FPmin}}{\Delta U_{Sensor}} = \frac{37,7 \text{ V} - 23,1 \text{ V}}{0,388 \text{ V}} = 37,629 \quad (3.5)$$

Somit kann bei der Verwendung von weniger als 37 Stufen über den gesamten Spannungsbereich des DC/DC Step-Up Wandlers die maximale spektrale Auflösung des Sensors nicht erreicht werden.

Zusätzlich gilt es, die Spannungsgenauigkeit der Steuerspannung zu betrachten. Die von Hamamatsu empfohlene Spannungsgenauigkeit der Steuerspannung des FPI beträgt  $\pm 60 \text{ mV}$  [13]. Real liegt die gemessene Restwelligkeit bei  $\pm 40 \text{ mV}$  und somit im Bereich der empfohlenen Toleranz. Mittels dieser für die Spannungserzeugung des FPI relevanten Resultate sind die für die Programmierung notwendigen Werte zur Nutzung der maximalen Auflösung des Sensors bekannt.

Um einen realen Wert der erzeugten Spannung zu erhalten, ist der Ausgang des DC/DC Step-Up Wandlers mit dem Eingang eines Analog-Digital-Converter (ADC)-Kanals verbunden (vgl. Abbildung 3.3). Aufgrund der hohen Steuerspannung besteht diese Verbindung laut Schaltplan in Anlage B aus einem Spannungsteiler mit dem Eingangs-zu-Ausgangsverhältnis von 12,236. Dieses Verhältnis ist bei der Ermittlung der tatsächlich erzeugten Spannung zu beachten.

### 3.1.2 Signalverstärker

Wie aus dem Schaltplan der Leiterplatte in Anlage B hervorgeht, folgen dem Ausgangssignal der Photodiode zwei Verstärkerstufen. Das analoge Stromsignal der Photodiode wird zuerst durch einen Transimpedanzverstärker in eine elektrische Spannung umgewandelt. Anschließend wird diese Spannung erneut verstärkt, so dass eine maximale Spannung von  $3,3 \text{ V}$  erreicht werden kann. Diese als  $U_{SD}$  definierte Spannung liegt am Eingang des 16-bit Sigma-Delta (SD)-ADC an.

Betrieben wird die Photodiode mit der Sperrspannung  $U_{Sperr} = 0,503 \text{ V}$ , welche als statische Grundspannung additiv in  $U_{SD}$  eingeht. Wiederum additiv zu dieser Span-

nung beeinflusst die Dunkelspannung  $U_{Dunkel}$  das Eingangssignal des SD-ADC. Diese entsteht durch den Dunkelstrom der Photodiode, der typischerweise zwischen 10 nA und maximal 100 nA liegt [20]. Während  $U_{Sperr}$  als konstant angenommen wird, ist  $U_{Dunkel}$  temperaturabhängig und hat dadurch einen dynamischen Einfluss auf das Messsignal. Wird als Definition der Signalgröße des auf die Photodiode einfallenden Lichts am Eingang des SD-ADC  $U_{Licht}$  angenommen und durch  $U_{SD}$  die vom SD-ADC umgewandelte Spannung beschrieben, ergibt sich die Beziehung:

$$U_{SD} = U_{Sperr} + U_{Dunkel} + U_{Licht} \quad (3.6)$$

Da lediglich  $U_{Licht}$  als Messwert relevant ist, sind die störenden Anteile aus  $U_{SD}$  zu eliminieren. Auf der Leiterplatte besteht dazu eine Verbindung zwischen dem negativen Differenzeingang des SD-ADC und einem Ausgang eines DAC, welcher eine Spannung  $U_{Ausgleich}$  erzeugt. Somit ergänzt sich Gleichung 3.6 zu:

$$U_{SD} = (U_{Sperr} - U_{Ausgleich}) + U_{Dunkel} + U_{Licht} \quad (3.7)$$

$U_{Ausgleich}$  wird über das Register DAC2DAT des 10-bit DAC eingestellt:

$$DAC2DAT = \frac{U_{Sperr} \cdot 2^{n_{bits}}}{U_{AVDD}} = \frac{0,503 \text{ V} \cdot 2^{10}}{3,3 \text{ V}} = 156,082 \quad (3.8)$$

Daraus folgt, dass der ganzzahlige Wert für DAC2DAT mit 156 zu belegen ist. Die für  $U_{Ausgleich}$  resultierende Spannung ergibt sich zu:

$$U_{Ausgleich} = \frac{U_{AVDD} \cdot DAC2DAT}{2^{n_{bits}}} = \frac{3,3 \text{ V} \cdot 156}{2^{10}} = 0,503 \text{ V} \quad (3.9)$$

Unter der getroffenen Voraussetzung, dass  $U_{Sperr} = U_{Ausgleich}$  entspricht, lässt sich Gleichung 3.7 vereinfachen:

$$U_{SD} = U_{Dunkel} + U_{Licht} \quad (3.10)$$

Somit ist der statische Einfluss von  $U_{Sperr}$  aus dem in digitale Form gewandelten Wert des SD-ADC eliminiert. Der temperaturabhängige Einfluss von  $U_{Dunkel}$  bleibt weiterhin bestehen. Weitere Erläuterungen dazu folgen in Abschnitt 3.3.

## 3.2 Auslegung des Messsystems

Die folgende Informationssammlung und Planung bezieht sich auf den verwendeten NIR MEMS-FPI Spektroskop. Dabei werden die wesentlichen für die Implementierung erforderlichen Abläufe und Berechnungen festgelegt.

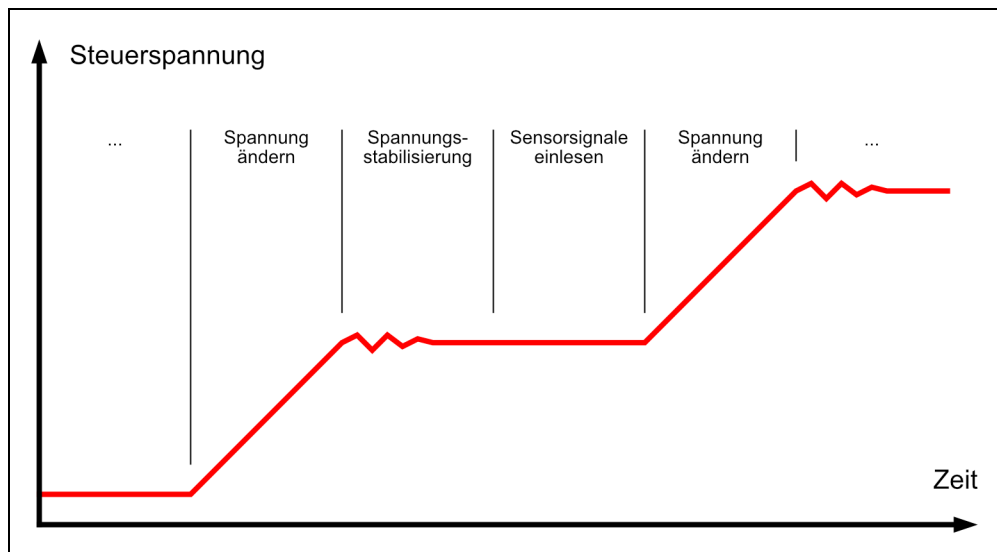
### 3.2.1 Steuerspannung

Für die Messung eines spektralen Bereiches wird die Steuerspannung des Sensors stufenweise verändert. Dabei wird bei jeder eingestellten Spannung das verstärkte Signal der Photodiode sowie das Signal des im Sensor integrierten Negative Temperature Coefficient Thermistor (NTC) eingelesen, welche sich zusammenfassend als Sensorsignale bezeichnen lassen. Weitere Erläuterungen zum NTC schließen in Abschnitt 3.2.3 an.

Wie in Abbildung 3.4 qualitativ dargestellt, folgen einer Änderung der Steuerspannung mehrere sequentielle Abläufe. Nach jeder Anpassung der Steuerspannung auf ein anderes Niveau muss sich diese stabilisieren, um eine präzise Ansteuerung des FPI zu erzielen. Die zur Stabilisierung der Spannung benötigte Zeit nach dem Sprung einer Spannungsstufe ist mit einem Oszilloskop aufgrund der geringen Signalwelligkeit kaum messbar. Daher ist eine variable Verzögerung vor dem Einlesen der Sensorsignale in der Firmware einzuplanen und durch Tests auf einen Wert einzustellen, welcher minimale Toleranzen unter minimaler Verzögerung ermöglicht. Nach dem Einlesen der Sensorsignale wird die Steuerspannung erneut geändert und der Ablauf wiederholt sich, bis der gesamte Spannungsbereich abgedeckt ist.

Der letztendliche Verlauf der Steuerspannung bei einer Messung lässt sich idealerweise als linear beschreiben. Zur Minimierung von Messrauschen ist die mehrfache Aufnahme des spektralen Bereiches mit dessen anschließender Mittelwertbildung einzuplanen. Dadurch ergeben sich unter Betrachtung einer gesamten Messung mehrere aufeinander folgende Rampen als Verlauf der Steuerspannung. In Abbildung 3.4 ist der Verlauf der Steuerspannung beispielhaft als positive Rampe dargestellt. Allerdings ist es notwendig, alle möglichen Formen der Spannungsänderung zu validieren, da mehrere Faktoren Einfluss auf die letztendliche Genauigkeit und Geschwindigkeit einer Messung haben. Dazu zeigt Abbildung 3.5 vier Varianten

qualitativer Spannungsverläufe für jeweils eine Messung mit zweifachem Durchlauf des Spannungsbereiches.

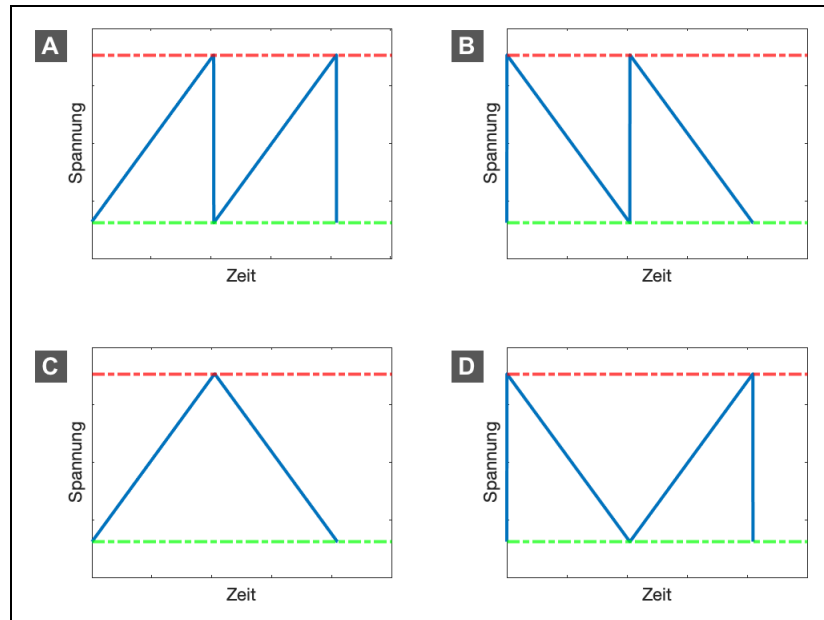


**Abb. 3.4:** Qualitativer Verlauf der Steuerspannung des Sensors. Dargestellt ist ein schematischer Ausschnitt von zwei Spannungsänderungen.

Für die gezeigten Verläufe A und B ergeben sich bei realer Betrachtung eindeutige Unterschiede. Ausgehend von der Steuerspannung  $U_{FP} = 0\text{ V}$  zu Beginn einer Messung benötigt der Sprung von  $U_{FP} = 0\text{ V}$  auf die maximale Steuerspannung  $U_{FPmax} = 37,7\text{ V}$  mehr als 40 ms (vgl. Anlage F). Hingegen ist die minimale Steuerspannung von  $U_{FPmin} = 23,1\text{ V}$  nach einem Sprung beginnend bei  $U_{FP} = 0\text{ V}$  bereits nach etwa 9 ms erreicht. Begründet werden kann dies durch die in den Aufzeichnungen sichtbare, kontinuierlich abnehmende Spannungsänderung pro Zeit, welche besonders bei großen Spannungssprüngen Einfluss hat. Obwohl sich dieser Werte nur auf die erstmalige Erzeugung der Steuerspannung ausgehend von  $U_{FP} = 0\text{ V}$  beziehen, ist eine deutliche Reduktion der Messdauer durch Nutzung der in Abbildung 3.5 A dargestellten Verlaufsform zu erreichen.

Unter Berücksichtigung dieser Ergebnisse erscheint die in Abbildung 3.5 C dargestellte Verlaufsform für die schnelle Aufnahme einer Messung vorteilhaft, da hier die Vorteile der Form A zu Tragen kommen und keine großen Spannungssprünge notwendig sind. Allerdings ist sowohl die Verlaufsform C, als auch Form D aufgrund der wechselnden Verlaufsrichtungen der Spannung und der dadurch möglichen Einflüsse von Hysterese ungeeignet für reproduzierbare Ergebnisse. Ein tatsächlicher Einfluss von Hysterese zwischen positiven Spannungssprüngen und

negativen Spannungssprüngen ist nicht bekannt. Durch das Ausschließen wechselnder Verlaufsformen ist diese allerdings in jedem Fall vermeidbar, weshalb eine Abwägung zwischen der Zeitdauer einer Messung und dem ungewissen Einfluss von Hysterese vorzunehmen ist.



**Abb. 3.5:** Qualitative Verläufe der Steuerspannung. Diese verläuft zwischen maximaler Steuerspannung (Rot) und minimaler Steuerspannung (Grün).

Für diese Arbeit ist aufgrund der Vernachlässigung anderer Faktoren, wie sie im weiteren Verlauf beschrieben werden, an dieser Stelle die Verlaufsform mit besserer Reproduzierbarkeit zu wählen. Somit ist von der Nutzung einer wechselnden Verlaufsform, wie sie in Abbildung 3.5 C und D dargestellt sind, abzusehen. Zusammenfassend überwiegen die Vorteile des in Abbildung 3.5 A dargestellten Verlaufes, welcher deshalb für die praktische Umsetzung anzuwenden ist.

Wie in Abschnitt 3.1.1 bereits erwähnt, sind der Spannungsbereich des Sensors und der Spannungsbereich der durch den DC/DC Step-Up Wandler erzeugbaren Spannung nicht identisch. Theoretisch ist es daher nicht notwendig, Sensorsignale für Steuerspannungen unter  $U_{\lambda 1850nm}$  einzulesen. Da das Pull-In Phänomen jedoch nicht bei Spannungen unter  $U_{\lambda 1850nm}$  auftritt und diese Spannung aufgrund ihrer Temperaturabhängigkeit nicht als dauerhaft konstant angenommen werden darf, sind die Sensorsignale über den gesamten Spannungsbereich des DC/DC Step-Up Wandlers einzulesen. Informationen, die sich außerhalb des spektralen Bereiches

des Sensors befinden, sind bei der weiterführenden Darstellung der Daten zu ignorieren.

### 3.2.2 Probenbelichtung

Die NIRS basiert auf der Belichtung der Probe (vgl. Abschnitt 2.1.1). Das mobile NIR-Spektrometer enthält acht Lampensockel, für welche vier verschiedene Typen von Halogenlampen zu Verfügung stehen. Halogenlampen eignen sich aufgrund des hohen Wirkungsgrades im sichtbaren Spektralbereich gut als Leuchtmittel für die optische Spektroskopie [25, S. 74]. Auch im an diesen Bereich angrenzenden NIR haben Halogenlampen eine hohe Lichtintensität [25, S. 206].

Die tatsächliche spektrale Verteilung des Lichts der für diese Arbeit bereitgestellten Lampen ist aufgrund fehlender Messtechnik nicht bekannt. Für die Auswahl eines Lampentyps ist daher ein Vergleich vorzunehmen.

Bei der direkten Einstrahlung auf den NIR MEMS-FPI Spektrolsensor, durch jeweils eine der vier Typen von Halogenlampen, ergeben sich die in Anlage G zusammengefassten Messdaten. Basierend auf diesen Daten bietet die Kombination aus vier Halogenlampen des Typs 7361 die meisten Vorteile. Mit einer Betriebsspannung von 3 V bei einem elektrischen Strom von 172 mA ergibt sich eine Signaldifferenz in Höhe von 128 mV zwischen beleuchtetem Sensor und nicht beleuchtetem Sensor. Somit ergibt sich im Vergleich die beste Kombination aus Stromverbrauch und Lichtintensität. Die gemessenen Werte gelten lediglich als Vergleichswerte zwischen den Lampentypen und dürfen nicht als Referenz für das letztendliche Ergebnis einer Messung des Spektrometers verwendet werden.

Da die reale spektrale Verteilung der Halogenlampe nicht bekannt ist, kann auch die im Vergleich zum Versuchsaufbau leicht höhere, gemessene Betriebsspannung  $U_{Lampen} = 3,14 \text{ V}$  auf der Leiterplatte für weitere Betrachtungen vernachlässigt werden. Zu beachten ist, dass die vom Spektrometer gemessene spektrale Verteilung einer Probe von der spektralen Verteilung der Halogenlampen abhängig ist und somit der quantitative Vergleich zwischen aufgenommenen Spektren einer Probe mit unterschiedlichen Leuchtmitteln ohne Anwendung von Chemometrie nicht möglich ist (vgl. Abschnitt 2.1.1).

Ergänzend bietet die Konzeption der Leiterplatte die Möglichkeit der Ansteuerung der Lampen mittels Pulsweitenmodulation (PWM) (vgl. Anlage B). Für das Ziel dieser Arbeit besteht keine Notwendigkeit, die Lampen nicht mit maximaler Lichtintensität zu betreiben, weshalb von dieser Option abzusehen ist.

### 3.2.3 Temperatureinfluss

Wie bereits für das FPI in Abschnitt 3.1.1 erwähnt, unterliegt der NIR MEMS-FPI Spektroskop einer Temperaturabhängigkeit. Diese entsteht durch die Charakteristiken der einzelnen Bauteile des Sensors in Hinsicht auf die Umgebungstemperatur, weshalb dieser über einen integrierten NTC verfügt [13]. Das „Final Inspection Sheet“ des in dieser Arbeit verwendeten Sensors definiert den Widerstand des NTC auf  $R_{NTC} = 9,949 \text{ k}\Omega$  bei  $25 \text{ }^\circ\text{C}$  Umgebungstemperatur (vgl. Anlage C).

Laut dem Hamamatsu „Instruction Manual“ ist die Beziehung zwischen Temperatur und NTC-Widerstand durch die Steinhart-Hart Gleichung beschrieben [13]. Zur Erhöhung der Genauigkeit wird empfohlen, den Widerstandsbereich aufzuteilen und das Verhältnis bereichsweise zu ermitteln. Aufgrund der fehlenden Möglichkeit weitere, reale Widerstandswerte des NTC zur Berechnung der notwendigen Konstanten zu ermitteln, werden die im Instruction Manual gegebenen Daten des Referenzbeispiels verwendet. Die Widerstandskurve wird anschließend an den tatsächlichen Widerstand  $R_{NTC}$  angepasst. Eine vollständige Berechnung der Beziehung zwischen Temperatur und NTC-Widerstand ist in Anlage H angefügt.

Die Umgebungstemperatur des Sensors wird neben der Raumtemperatur maßgeblich von den Halogenlampen beeinflusst, welche ringförmig um den Sensor angeordnet sind. Abbildung 3.6 zeigt die idealisierten Ergebnisse einer Messreihe der Temperatur des Sensors aufgetragen über die Zeit. Dabei sind die Halogenlampen ab dem Beginn der Messung eingeschaltet. Ausgehend von der gemessenen Temperatur von  $25 \text{ }^\circ\text{C}$  bei ausgeschalteten Lampen zeigt sich die Sättigung der Temperatur bei  $35 \text{ }^\circ\text{C}$  und somit ein durch die Lampen bedingter Temperaturanstieg von  $\Delta T = 10 \text{ K}$ .

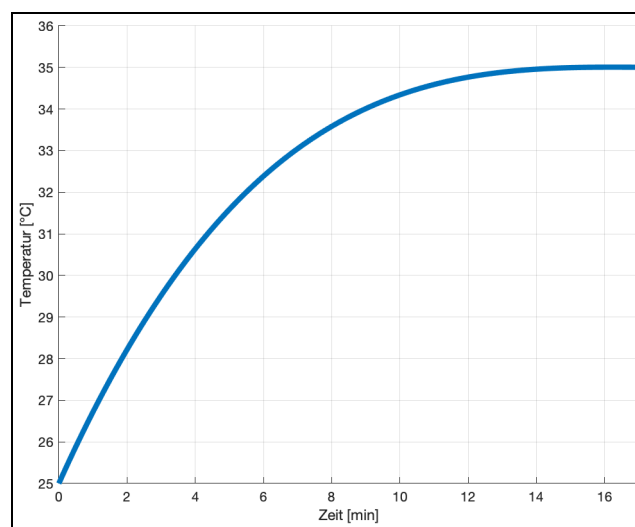
Für den in dieser Arbeit verwendeten Sensor ist eine typische Temperaturabhängigkeit von  $\Delta\lambda/T = 0,4 \text{ nm}/^\circ\text{C}$  bei  $\lambda = 1700 \text{ nm}$  angegeben [13]. Dadurch kann für den Betrieb mit eingeschalteten Lampen näherungsweise folgende Änderung der



Wellenlänge angenommen werden:

$$\Delta\lambda_T = \Delta\frac{\lambda}{T} \cdot \Delta T = 0.4 \frac{\text{nm}}{\text{K}} \cdot 10 \text{ K} = 4 \text{ nm} \quad (3.11)$$

Im Vergleich zur maximalen spektralen Auflösung des Sensors von  $\Delta\lambda_{FWHM} = 10 \text{ nm}$  (vgl. Abschnitt 3.1.1) ist daher der Temperatureinfluss in dieser Arbeit, unter dem bedingten Betrieb bei einer Raumtemperatur im Bereich von  $25 \text{ }^\circ\text{C}$ , zu vernachlässigen. Dennoch ist für eine mögliche Auswertung der Messdaten der Temperaturverlauf während einer Messung zu erfassen.



**Abb. 3.6:** Idealisierter Temperaturverlauf bei eingeschalteten Halogenlampen, gemessen durch den im Sensor integrierten NTC. Verwendet wurden vier Lampen des Typs 7361 bei  $U_{Lampen} = 3,14 \text{ V}$ .

Aufgrund der Temperaturabhängigkeit des Sensors ist die Empfehlung auszusprechen, jegliche Messungen erst 15 Minuten nach der Inbetriebnahme des NIR-Spektrometers vorzunehmen, da ab diesem Zeitpunkt die Temperatur des Sensors als konstant anzunehmen ist (vgl. Abbildung 3.6). Somit sind temperaturbedingte Einflüsse auf ein Minimum zu reduzieren und vergleichbare Messergebnisse erzielbar.

### 3.2.4 Wellenlängenberechnung

Die Beziehung zwischen Steuerspannung des Sensors und der zugehörigen Wellenlänge des auf die Photodiode einfallenden Lichts wird in dem Instruction Ma-

nual des Hamamatsu Sensors sowohl mit Temperaturkompensation, als auch ausgehend von einer Umgebungstemperatur von 25°C angegeben (vgl. [13]). Beide Varianten werden durch Polynomfunktionen hohen Grades beschrieben, welche zur Berechnung sensor-spezifischer Koeffizienten erfordern. Diese sind in den Tabellen des „Reference Data Sheet“ in Anlage D ersichtlich. Wie in dem vorherigen Abschnitt erläutert, ist der Temperatureinfluss für diese Arbeit zu vernachlässigen, weshalb kein weiterer Bezug auf dessen mathematische Kompensation genommen wird.

Für die Beziehung zwischen Steuerspannung und Wellenlänge ohne Temperaturkompensation ist die Funktion mit  $V$  als Steuerspannung,  $\lambda_p$  als Wellenlänge und den sensorspezifischen Koeffizienten  $a_0 - f_0$  definiert:

$$V^2 = a_0\lambda_p^5 + b_0\lambda_p^4 + c_0\lambda_p^3 + d_0\lambda_p^2 + e_0\lambda_p + f_0 \quad (3.12)$$

Ausgehend von der Steuerspannung als bekannte Variable besteht für diese Funktion keine allgemeine Lösungsform. Aus diesem Grund gibt Hamamatsu im Instruction Manual eine weitere, approximierte Lösungsform an (vgl. [13]). Diese wird vom weitgehend proprietären Evaluations-Kit C13294-02 genutzt und ermöglicht die Berechnung der Wellenlänge  $\lambda$  aus gegebener Steuerspannung  $V$  mittels acht weiteren sensor-spezifischer Koeffizienten  $a_0 - a_7$ :

$$\lambda = a_0 + a_1V + a_2V^2 + a_3V^3 + a_4V^4 + a_5V^5 + a_6V^6 + a_7V^7 \quad (3.13)$$

Obwohl die Umrechnung der Steuerspannung in die Wellenlänge aufgrund der Approximation keine komplexe Rechenaufgabe darstellt, ist diese Berechnung nicht auf dem Mikrocontroller auszuführen, sondern auf das Endgerät mit Visualisierungssoftware auszulagern. Begründet ist dies durch den geringen verfügbaren flüchtigen Speicher des Mikrocontrollers sowie das Fehlen einer Floating-Point Unit (FPU) zur präzisen Berechnung [12]. Obwohl eine alternative Berechnung mit Festkomma-Arithmetik möglich ist, ist damit die im Referenzbeispiel von Hamamatsu angegebene Approximationsgenauigkeit von  $\pm 0,1$  nm unter Umständen nicht zu erreichen (vgl. [13]).

### 3.3 Messdatenkonvertierung

Zur letztendlichen Darstellung der spektralen Verteilung einer Probe in der Visualisierungssoftware durch einen Graphen der in Abschnitt 2.1.1 beschriebenen Form sind neben der eigentlichen Messung der Probe die Daten von zwei weiteren Messungen erforderlich, die sich als sogenannte Referenzmessung zusammenfassen lassen. Eine Referenzmessung ist unter Verwendung einer Referenzprobe vorzunehmen und inkludiert jeweils eine Messung mit Belichtung und eine Messung ohne Belichtung. Als Referenzprobe eignet sich weißes Papier, da dies die notwendige Bedingung der vollständigen Reflexion nahezu erfüllt (vgl. [25] Abschnitt 4.7.2). Während die Referenzmessung mit Belichtung näherungsweise die spektrale Verteilung der verwendeten Lampen widerspiegelt, welche für die Berechnung der Extinktion relevant ist, dient die Referenzmessung ohne Belichtung dem Eliminieren des Einflusses von  $U_{Dunkel}$  auf  $U_{SD}$  (vgl. Abschnitt 3.1.2).

Die in Abschnitt 2.1.1 definierte Extinktion kann aufgrund der linear proportionalen Beziehung zwischen Intensität und dem in Spannung gewandelten Ausgangssignal einer Photodiode (vgl. [23] Abschnitt 5.4) als Funktion zweier Spannungen angegeben werden. Dabei bleibt die Bedeutung dieser Signale der ursprünglichen Definition erhalten:

$$A = \log_{10} \frac{U_0}{U} \quad (3.14)$$

Hierbei ergibt sich die Notwendigkeit der Referenzmessung mit Belichtung, da zur Berechnung der Extinktion die Intensität beziehungsweise die daraus resultierende Spannung der Belichtung erforderlich ist. Aufgrund der bereits in Abschnitt 3.2.2 erwähnten, nicht bekannten spektralen Verteilung der Leuchtmittel, ist diese unter Annahme der vollständigen Reflexion der Referenzprobe zu ermitteln. Zur korrekten Berechnung der Extinktion ist weiterhin der Einfluss von  $U_{Dunkel}$  zu eliminieren, indem die Messung der Referenzprobe ohne Belichtung idealerweise in  $U_{SD} = U_{Dunkel}$  resultiert. Formal ist dies unter Verwendung von  $U_{SD_{Probe}}$  für  $U_{SD}$  bei der Messung einer Probe,  $U_{SD_{RefON}}$  für  $U_{SD}$  bei einer Referenzmessung mit Belichtung und  $U_{SD_{RefOFF}}$  für  $U_{SD}$  bei einer Referenzmessung ohne Belichtung zusammengefasst:

$$A(\lambda) = \log_{10} \frac{U_{SD_{RefON}}(\lambda) - U_{SD_{RefOFF}}(\lambda)}{U_{SD_{Probe}}(\lambda) - U_{SD_{RefOFF}}(\lambda)} \quad (3.15)$$

Die weitere Auflösung der Gleichung ergibt auf Basis der in Abschnitt 3.1.2 verwendeten Symbole sowie der Definition von  $U_{Licht_{Ref}}$  als äquivalent zu  $U_{Licht}$  bei einer Referenzmessung:

$$A(\lambda) = \log_{10} \frac{U_{Dunkel}(\lambda) + U_{Licht_{Ref}}(\lambda) - U_{Dunkel}(\lambda)}{U_{Dunkel}(\lambda) + U_{Licht}(\lambda) - U_{Dunkel}(\lambda)} = \log_{10} \frac{U_{Licht_{Ref}}(\lambda)}{U_{Licht}(\lambda)} \quad (3.16)$$

Somit ist bewiesen, dass jegliche Einflüsse auf die Messung durch Anwendung einer Referenzmessung entfernt sind.

Unter Berücksichtigung der in Abschnitt 3.2.3 beschriebenen Empfehlung, Messungen erst 15 Minuten nach Inbetriebnahme des NIR-Spektrometers vorzunehmen, ist lediglich eine Referenzmessung für eine Reihe von Messungen von Proben notwendig. Daher ist es nicht erforderlich, eine Referenzmessung vor jeder Messung einer Probe vorzunehmen.

Für die Darstellung der Daten in der Visualisierungssoftware sind neben der graphischen Auftragung der Extinktion gegenüber der Wellenlänge weitere Formate darzustellen. Dazu zählen die vom SD-ADC eingelesenen digitalen Werte als Rohdaten und die Darstellung als Transmission gegenüber der Wellenlänge (vgl. Abschnitt 2.1.1). Um einen graphischen Vergleich mehrerer Messungen zu erhalten, sind zwei Messungen von Proben sowie die in der Referenzmessung inkludierten Messungen in einem Graphen anzuzeigen.

## 4 Design und Implementierung

Auf Basis der in der vorhergehenden Planung festgelegten Bedingungen werden im Folgenden das Design der Firmware und der Visualisierungssoftware erläutert und relevante Aspekte der Implementierung in generischer Form beschrieben. Firmware und Visualisierungssoftware haben in Anbetracht des Gesamtsystems unterschiedliche Aufgaben, teilen allerdings eine einheitliche Kommunikationsschnittstelle. Die Festlegung des Protokolls beziehungsweise des Datenformates, in dem die notwendigen Informationen ausgetauscht werden, ist wesentlich von dem Mikrocontroller beeinflusst, da dieser im Vergleich der Leistungsdaten den Flaschenhals darstellt. Aus diesem Grund wird das Kommunikationsprotokoll in Abschnitt 4.1.6.1 detailliert beschrieben und dessen Format in Abschnitt 4.2.2 für die Visualisierungssoftware als Vorgabe angenommen.

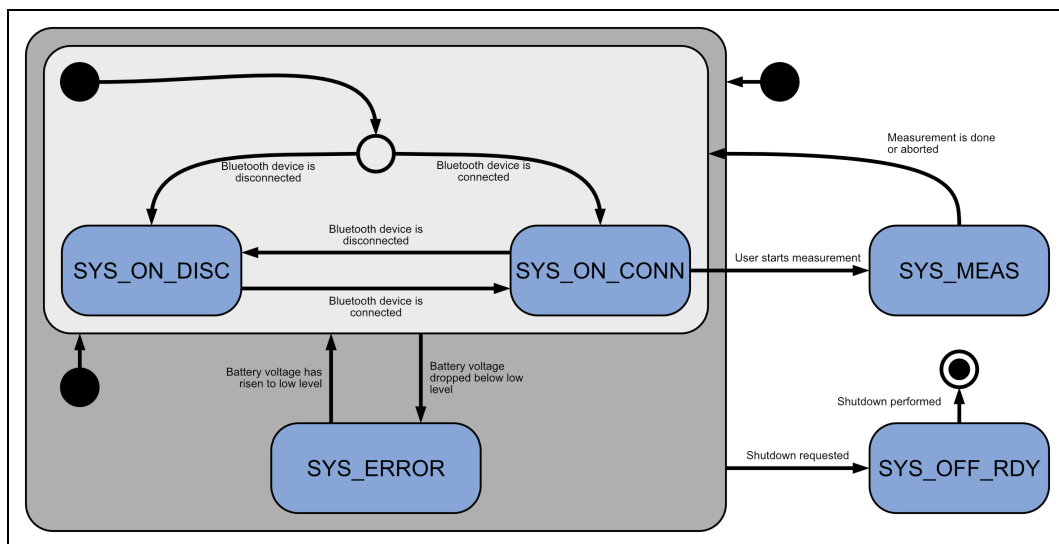
### 4.1 Firmware

Für die Firmware sind die Rahmenbedingungen des Mikrocontrollers zu betrachten. Dessen Taktfrequenz von 16 MHz wird durch den internen Oszillator mit anschließender Phasenregelschleife erzeugt und betreibt sowohl die Central Processing Unit (CPU), als auch die Peripherie. Die vom Mikrocontroller maximal unterstützte Taktfrequenz von 32 MHz (vgl. [12]) ist für den Anwendungsfall nicht notwendig und findet daher zu Gunsten der Effizienz keine Anwendung.

Große Relevanz für die Entwicklung der Firmware stellt zusätzlich der verfügbare Speicherplatz von flüchtigem Speicher in Form von Static Random-Access Memory (SRAM) und nicht-flüchtigem Flash-Speicher dar. Der Mikrocontroller enthält 8 KiB SRAM und 128 KiB Flash-Speicher (vgl. [12]), wodurch unter anderem der Verzicht auf die Nutzung eines Echtzeitbetriebsystems begründet ist. Zum einen ist dieses für die Anforderungen der Firmware nicht notwendig, zum anderen kann somit von der Nutzung von zusätzlich benötigtem Speicher und Rechenleistung abgesehen werden.

### 4.1.1 Programmstruktur

Wesentliche Aufgabe der Firmware ist das Sammeln und Bereitstellen von Informationen, die für die Darstellung der spektralen Verteilung einer Probe in der Visualisierungssoftware notwendig sind. Dazu sind nahezu alle Abläufe innerhalb der Firmware durch Zustandsautomaten gesteuert. Die grundlegende Steuerung des Systems folgt dem in Abbildung 4.1 dargestellten hierarchischen Zustandsautomaten mit fünf Zuständen. Im Folgenden werden diese Zustände, dessen Bezeichnungen denen im Quelltext entsprechen, als Systemzustände bezeichnet, da die aktiven Prozesse maßgeblich durch diese festgelegt sind.



**Abb. 4.1:** Zustandsdiagramm des hierarchischen Zustandsautomaten der Systemzustände. Die Zustandsbezeichnungen (blau) entsprechen den im Quelltext verwendeten.

Beim Einschalten des Gerätes wird der Systemzustand entsprechend dem Status der Bluetooth-Verbindung zur Visualisierungssoftware eingenommen. Solange keine Verbindung hergestellt ist und somit SYS\_ON\_DISC der aktive Systemzustand ist, kann keine Messung gestartet und der Systemzustand SYS\_MEAS nicht eingenommen werden. Unabhängig von dem Verbindungsstatus führt eine zu geringe Batteriespannung bei inaktiver Messung jederzeit zum Wechsel in den Fehlerzustand SYS\_ERROR. Die Relevanz dieser Funktion wird in Abschnitt 4.1.3 detailliert beschrieben.

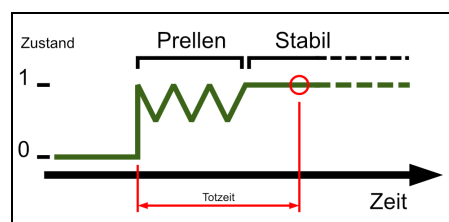
Das Ausschalten des Gerätes durch Aktivierung des Systemzustands SYS\_OFF\_RDY ist in der Anwendung jederzeit möglich, obwohl sich die Zustände SYS\_MEAS und

SYS\_OFF\_RDY ohne direkte Verbindung auf derselben Hierarchieebene befinden. Grund dafür ist die in Abschnitt 3.1 aufgezeigte Funktionsweise des Tasters K1 und dessen Belegung, wodurch eine aktive Messung stets abgebrochen und der Systemzustand SYS\_MEAS verlassen wird, bevor SYS\_OFF\_RDY eingenommen wird.

Obwohl der beschriebene Zustandsautomat die Basis des Programmablaufes bildet, sind parallele Prozesse aktiv, welche Einfluss auf die Änderungen der Systemzustände erlauben. Darunter ist zum Beispiel das Empfangen und Auswerten von Befehlen aus der Kommunikation mit der Visualisierungssoftware zu verstehen, welches in Abschnitt 4.1.6 näher beschrieben wird.

## 4.1.2 Benutzerschnittstelle

Die Bedienung des mobilen NIR-Spektrometers am Gerät selbst erfolgt lediglich durch den Taster K1 (vgl. Anlage A), welcher vier Funktionen kombiniert. Neben dem Einschalten und Ausschalten des Gerätes dient dieser ebenfalls zum Starten und Abbrechen einer Messung. Ausgehend vom ausgeschalteten Gerät wird dieses bei einfacher Betätigung des Tasters eingeschaltet (vgl. Abschnitt 3.1). Bei aktiver Bluetooth-Verbindung und wiederum einfacher Betätigung des Tasters wird der Systemzustand SYS\_MEAS eingenommen und die Messung gestartet. Solange eine Messung aktiv ist, wird diese durch Betätigung des Tasters K1 abgebrochen. Ausgeschaltet wird das Gerät nach dem dauerhaften Betätigen des Tasters für mindestens zwei Sekunden. Dem Anwender wird die Bereitschaft zum Ausschalten, welche dem Systemzustand SYS\_OFF\_RDY entspricht, durch die dauerhaft rot leuchtende RGB-LED signalisiert. Da der Taster K2 schaltungsbedingt keinen hardwareseitigen Reset des Mikrocontrollers erlaubt, wird dieser für die in der Entwicklung relevante Funktion zum manuellen Steuern der Lampen verwendet.



**Abb. 4.2:** Qualitativer Verlauf des Signals eines Taster bei Betätigung. Der stabile Signalzustand wird nach Ablauf der Totzeit abgefragt.

Obwohl die beiden im Gerät verbauten Taster bereits hardwareseitig entprellt sind (vgl. Abschnitt 3.1), werden diese zusätzlich in der Software entprellt, um maximale Störungsfreiheit zu erreichen. Dazu erzeugt jede Signalflanke eines Tasters einen Interrupt, in dessen Interrupt Service Routine (ISR) ein Flag (zu Deutsch: Merker) sowie ein Zeitstempel gesetzt werden. Dieser basiert auf einem Timer mit 1 ms Intervall und dessen inkrementierender Variable, welche dementsprechend als aktuelle Systemzeit in Millisekunden zu verstehen ist. Mit dem Flag wird im Programmablauf bei erkannter Signalflanke ein Vergleich zwischen der Addition von dem Zeitstempel und einer Totzeit von 5 ms mit der aktuellen Systemzeit vorgenommen. Abbildung 4.2 zeigt die qualitative Detektion einer Tasteneingabe unter Verwendung der softwareseitigen Entprellung und den damit verbundenen Hintergrund der Totzeit. Bei entsprechend abgelaufener Totzeit stellt das vom Taster eingelesene Signal den entprellten und aktuellen Zustand dar. Neben der Bedienung direkt am Gerät sind die genannten Funktionen sowie weitere Parameter über die Visualisierungssoftware steuerbar. Die genaue Beschreibung der Bedienung des Gesamtsystems wird zum einen sukzessive aus den Erläuterungen des Designs von sowohl Firmware als auch Visualisierungssoftware ersichtlich, zum anderen ist diese in der Kurzanleitung in Anlage J zusammengefasst.

Farbe	Intervall	Beschreibung
Rot	Permanent	Bereit zum Abschalten
Rot	4 Hz Blinken	Batterie schwach
Grün	Permanent	Messung aktiv
Grün	4 Hz Blinken	Referenzmessung aktiv
Orange	Permanent	Verbunden, bereit zur Messung
Orange	4 Hz Blinken	Verbindung getrennt

**Tab. 4.1:** RGB-LED Signale.

Die dem Anwender durch die RGB-LED angezeigten Informationen repräsentieren bis auf eine Ausnahme die Systemzustände und sind aus den drei Grundfarben Rot, Grün und Blau durch permanentes Leuchten oder Blinken codiert. Eine aktive Messung im Systemzustand `SYS_MEAS` differenziert sich in der visuellen Darstellung durch grünes permanentes Leuchten bei der Messung einer Probe und das grüne Blinken der LED bei einer aktiven Referenzmessung. Tabelle 4.1 listet die Codie-



rungen mit der jeweils zugehörigen Beschreibung der entsprechenden Information auf.

Um dem Anwender stets eine visuelle Rückmeldung über den aktuellen Status des Gerätes bei gleichzeitig effizienter Implementierung zu geben, erfolgt die Steuerung der RGB-LED basierend auf einem Timer mit 250 ms Intervall. Dadurch ergibt sich die in Tabelle 4.1 beschriebene Blinkfrequenz der RGB-LED von 4 Hz. In dem Programmablauf ist der gewünschte Status der RGB-LED lediglich einer Funktion zu übergeben, welche in Kombination mit dem in der ISR des Timers aufgerufenen Funktion die Kontrolle der Lampe zugehörigen GPIO-Pins automatisiert.

### 4.1.3 Batteriespannung

Für eine aktive Messung einer spektralen Verteilung sind die Ressourcen des Mikrocontrollers vollwertig für diese zur Verfügung zu stellen. Das dauerhafte Einlesen der Batteriespannung in einem fünf-Sekunden-Intervall erfolgt daher nicht im aktiven Systemzustand SYS\_MEAS.

Durch einen Spannungsteiler ist die am ADC anliegende Spannung der Batteriespannung gegenüber halbiert (vgl. Anlage B). Somit ist die Beziehung zwischen der Batteriespannung und dem digitalen Wert des 12 Bit-ADC definiert:

$$\frac{U_{Bat}}{2} = ADC \cdot \frac{U_{AVDD}}{2^{n_{Bits}}} = ADC \cdot \frac{3,3 \text{ V}}{2^{12 \text{ Bit}}} \quad (4.1)$$

Ziel ist die Darstellung der Batteriespannung als dezimale Festkommazahl mit einer Nachkommastelle. Dabei erfolgt durch Festkommaarithmetik die präzise Umrechnung des ADC-Wertes ohne FPU. Zur Vermeidung der Divisionsoperation und unter Verwendung einer vorzeichenlosen 32 Bit-Variable in der Implementierung ist zum Erhalt eines präzisen Ergebnisses das logische Verschieben des ADC-Wertes nach links vorzunehmen. Dazu ist die maximale Anzahl an logischen Bitverschiebungen  $x$  zu ermitteln, mittels welcher der temporäre Wert innerhalb des Wertebereiches der 32 Bit-Variable ist:

$$2^{32} < 20 \cdot ADC_{max} \cdot \frac{U_{AVDD}}{2^{n_{Bits}}} \cdot 2^x = 20 \cdot 4095 \cdot \frac{3,3 \text{ V}}{2^{12 \text{ Bit}}} \cdot 2^x \quad (4.2)$$

Durch Lösen der Gleichung ergibt sich für  $x$  ein Wert von 25,956 und somit eine

logische Verschiebung um 25 Bit innerhalb des Darstellungsbereiches einer 32 Bit-Variable. Für die effiziente Implementierung ist der Faktor  $Fac_{Bat}$  zu berechnen, welcher mit dem digitalen Wert des ADC multipliziert wird:

$$Fac_{Bat} = 20 \cdot \frac{3,3 \text{ V}}{4096} \cdot 2^{25 \text{ Bit}} = 540672 \quad (4.3)$$

Aufgrund des bereits in Gleichung 4.2 inkludierten Faktors 10 zum Erhalt einer Nachkommastelle ist anschließend lediglich die logische Verschiebung um 25 Bit nach rechts vorzunehmen:

$$U_{Bat} \cdot 10 = \frac{ADC \cdot Fac_{Bat}}{2^{25 \text{ Bit}}} \quad (4.4)$$

Das Ergebnis ist die faktorisierte Batteriespannung, welche in der ganzzahligen Form effizient in der Firmware behandelt und durch Division mit dem Faktor 10 als Gleitkommazahl in der Visualisierungssoftware weiter verwendet wird.

In Bezug auf das Gesamtsystem ist die Batteriespannung ein relevanter Faktor. Wie bereits in Abschnitt 4.1.1 beschrieben, wird das System bei einer zu geringen Batteriespannung in den Systemzustand `SYS_ERROR` versetzt, wodurch der Start einer Messung blockiert ist. Dies ist durch die minimal notwendige Versorgungsspannung des DC/DC Step-Up Wandlers zur Erzeugung der Steuerspannung begründet, welche auf Basis von Tests auf 3,5 V festgelegt ist. Somit hat die Batteriespannung einen steuernden Einfluss auf die Firmware und ist als ein paralleler Prozess zu dem in Abbildung 4.1 dargestellten Zustandsautomaten zu verstehen.

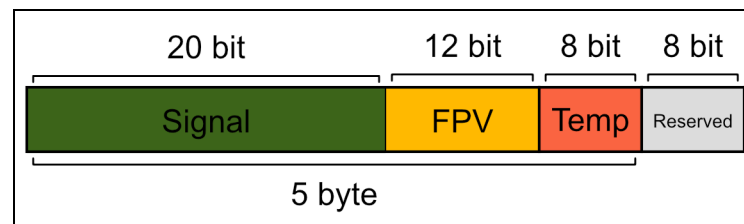
#### 4.1.4 Spektralsensor

Aufgrund des kleinen SRAM des Mikrocontrollers von 8 KiB (vgl. [12]) ist für die temporäre Speicherung der Daten einer Messung deren Speicherbedarf zu prüfen. Der minimal benötigte Speicherplatz einer Messung ist unter Verwendung der maximal 1024 Spannungsstufen der Steuerspannung des FPI (vgl. Abschnitt 3.1.1) und den einzulesenden Datenmengen zu berechnen. Jede Spannungsstufe, bei der Daten eingelesen werden, wird als Sample Point bezeichnet. Somit entsteht bei jedem Sample Point eine zu speichernde Datenmenge, die sich aus dem 16 Bit-Wert des SD-ADC für das Signal der Photodiode sowie zwei 12 Bit-Werten der ADC für

die tatsächliche Steuerspannung und das Signal des NTC bildet. Bedingt durch die Architektur des Mikrocontrollers (vgl. Abschnitt 3.1) und die damit verbundenen Eigenschaften der Implementierung in Strukturen ist die Datenmenge eines Sample Points auf Vielfache von 2 Byte beschränkt. Unter Beachtung der für die erfassten Daten benötigten 5 Byte ist für jeden Sample Point zusätzlich 1 Byte für zukünftige Ergänzungen reserviert, wodurch sich bei der maximalen Speicherung von 1024 Sample Points die Datenmenge einer Messung ergibt:

$$\text{Measurement Datasize} = 1024 \cdot 6 \text{ Byte} = 6144 \text{ Byte} \quad (4.5)$$

Die ermittelte Datenmenge ist kleiner als die im Mikrocontroller verfügbaren 8 KiB SRAM (vgl. [12]), weshalb die temporäre Speicherung aller Messdaten möglich und zusätzlich genügend Speicher für den Ablauf der Anwendung verfügbar ist. Im Folgenden ist für die Datenmenge eines Sample Points zur Erhaltung des Reserve-Bytes stets 5 Byte anzunehmen.



**Abb. 4.3:** Daten eines Sample Points. Die eingelesene Steuerspannung des FPI (FPV), das Signal der Photodiode (Signal) und der Temperaturwert (Temp) benötigen 5 Byte Speicher pro Sample Point.

Für die Anordnung der Daten eines Sample Points im Speicher ist die in Abschnitt 3.2.1 beschriebene Mittelwertbildung der Signale der Photodiode aus mehreren Aufnahmen zu beachten. Ziel ist, die in Gleichung 4.5 errechnete Datenmenge einer Messung beizubehalten, ohne relevante Information zu verlieren. Abbildung 4.3 zeigt die gewählte Aufteilung der Daten eines Sample Points. Während für den digitalen Wert der Steuerspannung der Ausgabe des ADC entsprechend 12 Bit zur Verfügung stehen, umfasst der Speicherplatz für den digitalen Wert des SD-ADC 20 Bit und ist somit 4 Bit größer als der Ergebniswert. Unter Beachtung der insgesamt für die Daten reservierten 5 Byte ist die Temperatur in 8 Bit zu speichern. Dabei orientiert sich die für den Temperaturwert verfügbare Datenmenge an der

einheitlichen Größe eines Bytes.

Für die Auslegung dieser Zuordnung sind die Prioritäten der einzelnen Daten zu betrachten. Zur Berechnung der Wellenlänge aus der Steuerspannung ist für eine präzise Zuordnung zur Wellenlänge die maximale Auflösung zu erhalten. Hingegen erfordert die Mittelwertbildung mehrerer Aufnahmen die Addition von 16 Bit-Werten und somit eine größere Speicherbreite. Weshalb die Minimierung der Datenmenge für den Temperaturwert keinen relevanten Informationsverlust mit sich bringt, wird in Abschnitt 4.1.4.3 detailliert erläutert.

Ausgehend von der für das Signal der Photodiode verfügbaren 20 Bit Speicherkapazität ist die maximale Anzahl an Aufnahmen zur Mittelwertbildung zu ermitteln. Dabei ist zu beachten, dass der SD-ADC einen 16 Bit-Wert als Ergebnis liefert, dieser aber das Ergebnis einer bipolaren Differenzspannung ist. Für die Anwendung sind lediglich positive Spannungen und somit eine Sättigung für den Bereich der positiven Zahlen anzuwenden. Die dadurch pro Sample Point notwendige Speicherkapazität für das Signal der Photodiode verringert sich somit auf 15 Bit. Daraus ergibt sich auf Basis der im Speicher verfügbaren 20 Bit die maximale Anzahl an Aufnahmen, welche durch deren Mittelwertbildung das letztendliche Messergebnis eines Sample Points bilden:

$$\text{Average Measurement Count} = \frac{2^{20\text{Bit}}}{2^{15\text{Bit}}} = 32 \quad (4.6)$$

Während die in den 20 Bit eines Sample Points gespeicherten Daten das arithmetische Mittel von der gewünschten Anzahl an Aufnahmen widerspiegelt, enthalten die zwei weiteren Werte lediglich die Informationen der letzten Aufnahme und stellen somit keinen Mittelwert dar.

Vorausschauend auf die in Abschnitt 4.1.6 folgende Auslegung der Kommunikation mit der Visualisierungssoftware ist für die Datenmenge einer Messung stets eine gerade Anzahl an Bytes beziehungsweise die Speicherausrichtung auf 16 Bit notwendig. Daraus ergibt sich die Wahl der Anzahl an Sample Points in Form von Zweierpotenzen. Ausgehend von maximal 1024 Sample Points und unter Beachtung der in Abschnitt 3.1.1 berechneten minimalen Anzahl von 37 Sample Points zur Einhaltung der maximalen Auflösung des Sensors ergeben sich die Werte 64, 128, 256, 512 und 1024. In Kombination mit der zuvor ermittelten maximalen

Anzahl an Aufnahmen zur Mittelwertbildung sind diese Werte ausschlaggebend für die Einstellungsmöglichkeiten, die dem Anwender auf Basis der technischen Möglichkeiten für eine Messung zur Verfügung stehen.

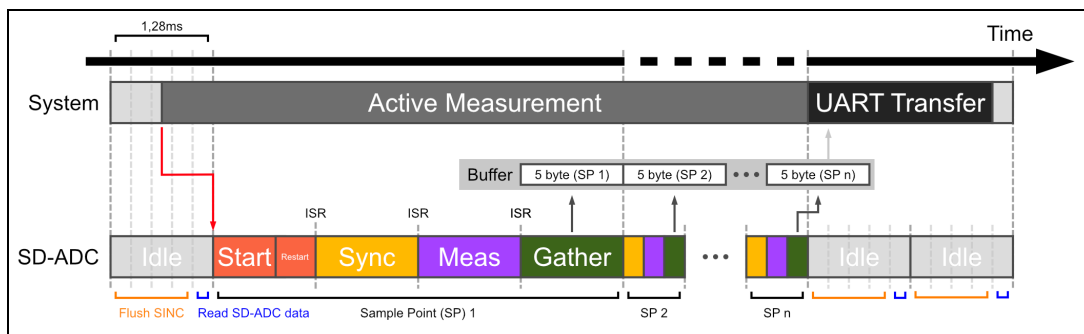
Der SD-ADC unterstützt lediglich einen kontinuierlichen Konvertierungsmodus (vgl. [8]), wodurch eine Synchronisation mit dem Programmablauf notwendig ist. Konfiguriert ist der SD-ADC mit einer Taktfrequenz von 4 MHz bei einer Oversampling-Ratio (zu Deutsch: Überabtastrate) von 256 und der daraus resultierenden Abtastrate von 3,906 25 ksp/s. Demnach ist alle 256  $\mu$ s ein digitaler Wert verfügbar. Um keine durch das interne SINC-Filter (zu Deutsch: ideale Tiefpassfilter) hervorgerufenen Signalverfälschungen zu erhalten, werden vier aufeinander folgende SD-ADC-Messungen verworfen und lediglich der Wert der fünften Messung verwendet. Relevant ist dies im Wesentlichen für den ersten Sample Point einer Messung, da das Filter vier Zyklen zum Einschwingen benötigt [8]. In der Anwendung ist dieses Verhalten allerdings für jeden Sample Point umgesetzt, sodass die Verfügbarkeit eines gültigen Wertes alle 1,28 ms durch einen Interrupt mitgeteilt wird.

Für diesen Anwendungsfall bietet die Peripherie des SD-ADC eine Option im Register SDxCON2, bei der lediglich jeder fünfte Interrupt zur Ausführung der ISR führt (vgl. [8]). Somit wird jeder fünfte eingelesene Wert vom SD-ADC an das Programm übergeben. Die weitere Konfiguration des SD-ADC ist entsprechend der Empfehlungen des Datenblattes.

Innerhalb der ISR werden die Abläufe durch einen Zustandsautomaten koordiniert, welcher nur im Systemzustand SYS\_MEAS aktiv ist und durch das Aufrufintervall der ISR zeitgesteuert arbeitet. Den Zustandsbezeichnungen „Start“, „Restart“, „Sync“, „Meas“ und „Gather“ entsprechend, beginnt eine Messung nach dessen Aktivierung im Programmablauf mit dem Zustand Start. Dieser initialisiert lediglich Variablen und wechselt direkt in den Zustand Restart, der das Erreichen des erforderlichen Spannungsniveaus der Steuerspannung prüft. Solange dieses zu Beginn einer Messung nicht erreicht ist, wird der Zustand inklusive Spannungsüberwachung wiederholend mit dem Intervall von 1,28 ms ausgeführt. Durch den im ADC integrierten Schwellwert-Vergleich wird bei der erreichten Steuerspannung von 23,1 V ein Flag gesetzt, welches in der ISR des SD-ADC abgefragt

wird. Aufgrund des in der Peripherie erfolgenden Schwellwert-Vergleiches ist der auszuführende Code innerhalb der ISR auf ein Minimum reduziert.

Bei Erreichen der Ziel-Steuerspannung wechselt der Zustandsautomat in den Zustand Sync. Dieser ermöglicht die in Abschnitt 3.2.1 als erforderlich deklarierte Stabilisierung der Steuerspannung, während die Messung der Signale verzögert wird. Für die Verzögerung zwischen jedem Sample Point ergibt sich, basierend auf der steuernden 8 Bit-Variable, der Bereich von 0 ms bis 326,4 ms. Während für diese Verzögerung in der letztendlichen Anwendung von sehr geringen Zeiten auszugehen ist, wird diese Funktion an Stelle der Referenzmessung mit einer größeren Verzögerung notwendig. Nähere Erläuterungen zu der notwendigen Verzögerung während einer Referenzmessung folgen in Abschnitt 4.1.5.



**Abb. 4.4:** Schematische Darstellung der Synchronisation des SD-ADC mit dem Programmablauf.

Nach Ablauf der Verzögerung wechselt der Zustandsautomat in den Zustand Meas, welcher zum einen die Konvertierung der beiden 12 Bit-ADC für die Erfassung der tatsächlichen Steuerspannung und dem Temperatursignal des NTC auslöst und zum anderen den Zeitbereich widerspiegelt, in dem der SD-ADC die Konvertierung vornimmt. Nach Ablauf des für 1,28 ms aktiven Zustands Meas wird im Zustand Gather entsprechend der Bezeichnung das Sammeln und Speichern der Daten in den Puffer vorgenommen. Dazu stehen pro Sample Point die zuvor beschriebenen 5 Byte zur Verfügung. Das detaillierte Format der jeweiligen Daten wird in den folgenden Abschnitten erläutert.

Anschließend wiederholt sich der gesamte Vorgang für den nächsten Sample Point, beginnend mit dem Zustand Sync, bis alle Sample Points erfasst sind. Bei der eingestellten mehrfachen Aufnahme mit anschließender Mittelwertbildung wird nach

dem letzten Sample Point einer Aufnahme der Zustand Restart eingenommen und die erneute Aufnahme gestartet. Die letztendliche Mittelwertbildung findet im Zustand Gather des letzten Sample Points statt. Aufgrund des zeitlichen Ablaufes des Zustandsautomaten ergibt sich das minimale Intervall zwischen zwei behandelten Sample Points von 3,84 ms.

Nach Abschluss der eigentlichen Messung werden die im Puffer gesammelten Daten über die UART-Schnittstelle an die Visualisierungssoftware übertragen. Die detaillierte Übertragung der Daten wird in Abschnitt 4.1.6.2 genauer beschrieben. Abbildung 4.4 fasst den zuvor beschriebenen Ablauf einer Messung und die Synchronisation mit dem Programmablauf in einer schematischen Darstellung zusammen.

#### 4.1.4.1 Wellenlängenfilter

Für die Verifikation der aktuellen Steuerspannung des im Sensor integrierten FPI wird diese wie bereits in Abschnitt 3.1.1 beschrieben über einen ADC eingelesen und als Basis für die Berechnung der gefilterten Wellenlänge verwendet. Somit ist gewährleistet, dass Abweichungen zwischen eingestellter Steuerspannung und der tatsächlich am FPI anliegenden Steuerspannung keinen Einfluss auf das Messergebnis haben. Ausgehend von dem zuvor beschriebenen verfügbaren Speicher für diese Messgröße erfolgt die Umrechnung des digitalen Wertes nicht in der Firmware, da diese aufgrund der identischen Speichergröße von eingelesener Datenmenge und verfügbarem Speicherplatz von 12 Bit zu der Reduktion der Signalaufösung führen würde. In Anbetracht der zu der Visualisierungssoftware übertragenen Daten ergibt sich die Beziehung zwischen realer Steuerspannung  $U_{FP}$  und digitalem Wert  $ADC_{FP}$  unter Berücksichtigung des in Abschnitt 3.1.1 definierten Faktors  $Fac_{FP}$  zwischen realer Steuerspannung und am ADC anliegender Spannung von 12,236:

$$ADC_{FP} = U_{FP} \cdot \frac{2^{n_{Bits}}}{U_{AVDD} \cdot Fac_{FP}} = U_{FP} \cdot \frac{2^{12 \text{ Bit}}}{3,3 \text{ V} \cdot 12,236} \quad (4.7)$$

Dementsprechend wird  $ADC_{FP}$  aus dem Ergebnisregister des ADC direkt in den entsprechenden Speicherbereich im Puffer kopiert, wobei die durch die elektronische Auslegung bedingte Faktorisierung des Wertes in der weiteren Datenverarbeitung in der Visualisierungssoftware zu berücksichtigen ist.

#### 4.1.4.2 Sensorsignal

Die grundlegende Auslegung des Speichers auf die vom SD-ADC eingelesenen Werte der Photodiode wurde bereits in Abschnitt 4.1.4 erläutert. Daher ist bei der weiteren Beschreibung des Datenformates von einem vorzeichenlosen 15 Bit-Wert auszugehen, welcher sich aus dem Ergebnisregister des SD-ADC unter Anwendung der Sättigung auf den positiven Zahlenbereich ergibt. Durch die entsprechende Konfiguration einer Messung wird das letztendliche Messergebnis aus bis zu 32 einzelnen Aufnahmen ermittelt (vgl. Abschnitt 4.1.4). Während einer Messung erfolgt sukzessive die Addition der vom SD-ADC konvertierten und gesättigten Werte von Sample Points gleicher Indizes.

Die abschließende Mittelwertbildung findet bei allen Sample Points der letzten Aufnahme unter Verwendung einer 32 Bit-Variable statt. Aufgrund der für die Berechnung zur Verfügung stehenden Speichergröße sowie des 20 Bit großen Speicherbereiches pro Sample Point im Puffer, ist für die Darstellung als dezimale Festkommazahl dessen maximale Anzahl an Nachkommastellen zu ermitteln. Dazu ist der größte Multiplikator des maximalen 12 Bit-Wertes zu berechnen, durch den der Zahlenbereich von 32 Bit nicht überschritten wird:

$$\frac{2^{20 \text{ Bit}}}{2^{12 \text{ Bit}}} = 256 \quad (4.8)$$

Daraus folgt die Darstellung als dezimale Festkommazahl mit zwei Nachkommastellen, da der digitale Wert mit dem als maximal berechneten dezimalen Faktor von 100 den Speicherbedarf eines 20 Bit-Wertes nicht übersteigt.

#### 4.1.4.3 Temperatursensor

Der im NIR MEMS-FPI Spektralsensor integrierte NTC bildet in Kombination mit einem Widerstand  $R_{13} = 10 \text{ k}\Omega$  einen Spannungsteiler (vgl. Anlage B). Dadurch ergibt sich bei der Betriebsspannung  $U_{AVDD}$  die am NTC abfallende Spannung  $U_{NTC}$  in Abhängigkeit dessen Widerstandes  $R_{NTC}$ :

$$U_{NTC} = \frac{U_{AVDD} \cdot R_{13}}{R_{13} + R_{NTC}} = \frac{3,3 \text{ V} \cdot 10 \text{ k}\Omega}{10 \text{ k}\Omega + R_{NTC}} \quad (4.9)$$



Diese Spannung wird vom ADC in den digitalen Wert  $ADC_{NTC}$  gewandelt und ist in dessen Abhängigkeit darstellbar:

$$U_{NTC} = ADC_{NTC} \cdot \frac{U_{AVDD}}{2^{n_{Bits}}} = ADC_{NTC} \cdot \frac{3,3 \text{ V}}{2^{12}} \quad (4.10)$$

Durch die Zusammenführung der Gleichungen 4.9 und 4.10 ergibt sich die Beziehung zwischen dem digitalen Wert  $ADC_{NTC}$  und dem Widerstand  $R_{NTC}$ :

$$R_{NTC} = \frac{-R_{13} \cdot ADC_{NTC}}{ADC_{NTC} - 2^{n_{Bits}}} = \frac{-10 \text{ k}\Omega \cdot ADC_{NTC}}{ADC_{NTC} - 4096} \quad (4.11)$$

Auf Basis dieser Beziehung ist mittels der Steinhart-Hart Gleichung (vgl. Abschnitt 3.2.3) die Temperatur aus dem digitalen Wert des ADC zu ermitteln.

Obwohl der ADC einen 12 Bit-Wert liefert, stehen für die Speicherung der Temperaturinformation eines Sample Points lediglich 8 Bit zur Verfügung (vgl. Abbildung 4.3). Bei einer Temporauflösung von  $1^\circ\text{C}$  ist für die Betriebstemperatur des Sensors zwischen  $-40^\circ\text{C}$  und  $+85^\circ\text{C}$  (vgl. [20]) der Darstellungsbereich von 130 Werten erforderlich. Ein vorzeichenbehafteter 8 Bit-Wert erlaubt mit einem Zahlenbereich von -128 bis +127 die direkte Darstellung der Temperaturen in Grad Celsius, weshalb die Umrechnung des digitalen ADC-Wertes in die Temperatur auf dem Mikrocontroller implementiert ist. Dafür ist ein Lookup Table (LUT) im Flash-Speicher des Mikrocontroller abgelegt, welcher die für den verwendeten Sensor spezifische Temperatur in Grad Celsius für jeden der 256 möglichen Digitalwerte enthält. Aufgrund der für die Temperaturdarstellung im Speicher verfügbaren 8 Bit ist keine Notwendigkeit eines LUT für die aus 12 Bit resultierenden 4096 Werte gegeben. Anlage H fasst die Berechnung dieser Werte und die daraus folgende Erzeugung des LUT zusammen.

#### 4.1.5 Probenbelichtung

Die Beleuchtung der Probe ist für die Erfassung einer spektralen Verteilung mit dem NIR MEMS-FPI Spektroskop unerlässlich. Um eine schnelle Temperaturstabilität des Systems zu erreichen, werden die Lampen automatisch eingeschaltet, sobald das mobile NIR-Spektrometer mit der Visualisierungssoftware verbunden ist. Während die Lampen dementsprechend nur in den Systemzuständen

SYS\_ON\_CONN und SYS\_MEAS eingeschaltet sind und somit durch diese gesteuert werden, ist die Aufnahme einer Referenzmessung zur Eliminierung des Dunkelstroms aus dem Signal (vgl. Abschnitt 3.3) eine Ausnahme, bei der die Lampen auch im Systemzustand SYS\_MEAS kurzzeitig ausgeschaltet sind.

Der technische Ablauf einer Referenzmessung entspricht grundlegend der schematischen Darstellung in Abbildung 4.4 und somit dem der Messung einer Probe. Allerdings wird nach dem Fertigstellen der Datenübertragung zur Visualisierungssoftware der Systemzustand SYS\_MEAS beibehalten und eine weitere Messung mit deaktivierter Beleuchtung gestartet. Dadurch erhält die Visualisierungssoftware die Daten von zwei einzelnen Messungen als Ergebnis einer Referenzmessung. Um das Abkühlen der Lampen zu ermöglichen und die damit verbundene Verteilung der Lichtintensität während einer Messung konstant zu halten, wird die in Abschnitt 4.1.4 beschriebene Verzögerung zwischen zwei Sample Points genutzt. Dabei wird in der Implementierung die Dauer der Verzögerung des ersten Sample Points durch einen statischen Wert ersetzt, welcher bei finalen Tests festzulegen ist. Nach Abschluss der Referenzmessung wird der Systemzustand SYS\_MEAS wieder verlassen, wodurch sich die Aktivität der Beleuchtung wieder dem aktuellen Systemzustand anpasst.

Für den Anwender ist lediglich die in Abschnitt 3.2.3 beschriebene Aufwärmzeit bis zur Temperatursättigung zu beachten. Eine manuelle Bedienung der Lampen ist nicht notwendig, allerdings durch Betätigung des Tasters K2 möglich (vgl. Abschnitt 4.1.2).

### **4.1.6 Datenübertragung**

In Anbetracht aller bisherigen Erläuterungen und Beschreibungen ergeben sich die für die Datenübertragung zwischen mobilem NIR-Spektrometer und Visualisierungssoftware relevanten Daten. Abgesehen von Steuerungsfunktionen und Informationen zu dem aktuellen Systemstatus sind vor allem die zur korrekten Darstellung einer Messung in der Visualisierungssoftware notwendigen Parameter zu übertragen. Darunter zählt die Anzahl an Sample Points, die Anzahl an Mittelwertbildungen, die Dauer der Verzögerung zur Stabilisierung der Steuerungspannung zwischen zwei Sample Points sowie die Information, ob die Messung

einer Referenzmessung entspricht.

Zu beachten ist, dass obwohl die Verzögerung zur Spannungsstabilisierung zwischen zwei Sample Points Teil der Datenübertragung ist, der entsprechende Wert statisch in der Firmware hinterlegt ist und nicht verändert werden kann. Allerdings sind die zugehörigen Programmzeilen zur dynamischen Änderung des Wertes in der Implementierung auskommentiert, wodurch diese Funktion bei Bedarf aktiviert werden kann. Ein weiterer Wert, welcher zur Identifikation der beiden Messungen innerhalb einer Referenzmessung dient, spiegelt den Status der Beleuchtung wieder. Die in insgesamt 8 Byte kombinierten Daten sind in Anlage I in detaillierter Form mit den entsprechenden Datenformaten strukturiert.

Des Weiteren erfordert die in Abschnitt 3.3 beschriebene Umrechnung der Steuerspannung in eine Wellenlänge sensorspezifische Koeffizienten, weshalb diese als zusätzliche Informationen zu einer Messung hinzugefügt sind. Die insgesamt 20 sensorspezifischen Koeffizienten sind in der Firmware als konstante 8 Byte-Binärwerte im Binärformat „binary64“ nach der Norm 754 des Institute of Electrical and Electronics Engineers (IEEE) (vgl. [27]) hinterlegt und entsprechen somit einer Datengröße von 160 Byte. Zusammenfassend vergrößert sich die in Gleichung 4.5 ermittelte maximale Datengröße einer Messung durch den Anhang der genannten Informationen:

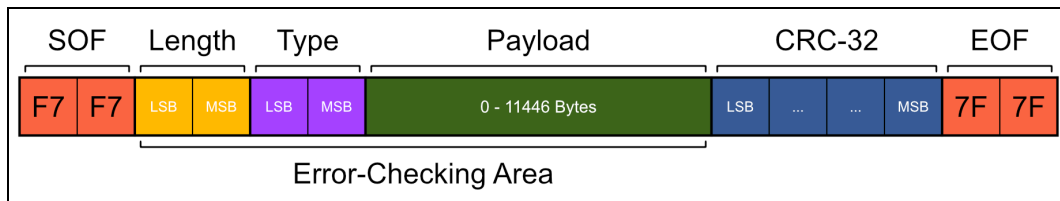
$$\textit{Extended Measurement Size} = 6144 \text{ Byte} + 160 \text{ Byte} + 8 \text{ Byte} = 6312 \text{ Byte} \quad (4.12)$$

Auf Grundlage dieser erweiterten Datenmenge basiert die im anschließenden Abschnitt erläuterte Auslegung des Kommunikationsprotokolls.

#### 4.1.6.1 Kommunikationsprotokoll

Die UART-Kommunikation überträgt einen Datenstrom, der aus einzelnen Bytes besteht. Für das Übertragen von Datenpaketen ist ein darauf aufbauendes Kommunikationsprotokoll erforderlich, welches aus den Anforderungen der vorhergegangenen Erläuterungen zu spezifizieren ist. Dazu zählt unter anderem die Festlegung des strukturellen Aufbaus eines Datenpaketes, welches zusammenfassend auch als Frame bezeichnet wird.

Abbildung 4.5 zeigt den aus den Anforderungen erstellten Aufbau eines Frames, welcher durch ein Start of Frame (SOF)-Symbol und ein End of Frame (EOF)-Symbol zu identifizieren ist. Basierend auf der 16 Bit-Architektur des Mikrocontrollers (vgl. [12]) ist die Ausrichtung und Größe der einzelnen Daten an Vielfachen von 2 Byte orientiert. Somit ist die Angabe der Datenmenge in der Payload (zu Deutsch: Nutzlast) im Feld „Length“ ebenfalls als Anzahl von 2 Byte-Worten zu verstehen.



**Abb. 4.5:** Aufbau eines Frames. SOF und EOF sind als konstante Hexadezimalwerte angegeben. Die Byte-Reihenfolge ist entsprechend dem Mikrocontroller *Little-endian*.

Basierend auf dem Ergebnis aus Gleichung 4.12 ergibt sich die erforderliche Datenmenge, welche in der Payload des Frames übertragen werden muss. Beschränkt ist die maximale Datenmenge der Payload durch das Feld „Length“, dessen 16 Bit unter Beachtung der Darstellung in 16 Bit-Worten eine theoretische Datenmenge von 131 070 Byte erlaubt. Entscheidend ist allerdings die Erhaltung der Datenintegrität durch den Cyclic Redundancy Check (CRC) und die damit verbundene maximale Datenmenge zur Einhaltung einer akzeptablen *Hamming-Distanz*. Aufgrund der Größe der Datenmenge ist ein CRC mit 32 Bit-Polynom anzuwenden, da unter Verwendung eines 16 Bit-Polynoms keine Datenintegrität über den gesamten Prüfbereich des CRC gewährleistet ist [1]. Tiefgreifende Analysen und Erläuterungen für die Wahl eines Generatorpolynoms sowie dessen genaue Charakteristik bezüglich der Fehlererkennung werden in dieser Arbeit nicht umfasst. Allerdings sind aufgrund der Verwendung eines CRC dessen Eigenschaften hinsichtlich der verwendeten Datenmengen zu überprüfen.

Unter Verwendung des 32 Bit Generatorpolynoms  $4C11DB7_{16}$  und dem Startwert  $FFFFFFF_{16}$  nach der Norm 802.3 des IEEE (vgl. [26]) ergibt sich für verwendete Datenmengen von 372 Byte bis 11 450 Byte eine minimale Hamming-Distanz von vier, wodurch bei maximaler Datenmenge einer Messung innerhalb eines Frames bis zu drei Bitfehler erkannt werden [22]. Bei kleineren Datenmengen ist die Hamming-

Distanz entsprechend größer, sodass bei einem Frame ohne Payload die von dem CRC abgedeckten 4 Byte eine minimale Hamming-Distanz von zehn erreichen. Zusammenfassend ergibt sich unter Beibehaltung der minimalen Hamming-Distanz von vier eine maximale Payload-Datenmenge von 11 446 Byte, da neben der Payload auch die Felder „Length“ und „Type“ in den CRC einbezogen werden.

Über das Feld „Type“ wird der Inhalt des Datenpaketes identifiziert. Dazu sind unter Betrachtung der für das Gesamtsystem relevanten Funktionen die in Tabelle 4.2 aufgelisteten Werte festgelegt. Der detaillierte Inhalt der Nachrichten ist in Anlage I zusammengefasst, wobei die in Tabelle 4.2 ersichtliche variable Größe der Payload zu beachten ist.

Type	Direction	Description	Payload Size (2 Byte-Word)
27 <sub>16</sub>	Send	Battery Voltage	1
63 <sub>16</sub>	Send	Measurement Data	276 - 3156
71 <sub>16</sub>	Receive	Stop Measurement	0
72 <sub>16</sub>	Receive	Start Measurement	0 (4)
73 <sub>16</sub>	Receive	Shutdown Device	0
77 <sub>16</sub>	Send	System Status Change	1
78 <sub>16</sub>	Receive	Settings	4

**Tab. 4.2:** Frame Type Codes. Die Richtung ist entsprechend des mobilen NIR-Spektrometers zu betrachten.

Die Kommunikation erfolgt dem Protokoll entsprechend verbindungslos und nicht-zuverlässig, wobei aufgrund der zugrundeliegenden UART-Schnittstelle Datenpakete immer in der gesendeten Reihenfolge empfangen werden. Für den gegebenen Anwendungsfall einer nicht-kommerziellen Lösung ist dies gegenüber der Sicherung der Datenintegrität allerdings nicht zu priorisieren und somit akzeptabel.

Das Protokoll erlaubt neben dem Informationsaustausch basierend auf Ereignissen auch die Übertragung von Daten auf Basis eines Zeitintervalls. Somit wird bei jedem Wechsel des Systemzustands eine Nachricht mit dem aktuellen Zustand an die Visualisierungssoftware gesendet, während der aktuelle Wert der Batteriespannung mit einem Zeitintervall von 5 s übertragen wird.

#### 4.1.6.2 Serielle Schnittstelle

Wie bereits in Abschnitt 3.1 beschrieben, kommuniziert der Mikrocontroller über die integrierte UART-Schnittstelle mit dem Bluetooth Modul. Dabei ist die UART-Schnittstelle des Bluetooth Moduls standardmäßig mit einer *Baudrate* von 115,2 kBd konfiguriert [10]. Weiterhin umfasst die Konfiguration acht Datenbits, ein Stoppbit, kein Paritätsbit und die deaktivierte Flusskontrolle. Für die Kommunikation mit dem Mikrocontroller sind die Einstellungen dessen UART-Peripherie entsprechend anzupassen.

Die Konfiguration der Baudrate erfolgt laut Datenblatt der im Mikrocontroller integrierten UART-Peripherie mit dessen Taktfrequenz  $F_P$  durch Setzen des Registers  $UxBRG$  entsprechend dem Ergebnis der Gleichung (vgl. [11]):

$$UxBRG = \frac{F_P}{4 * \text{Baud Rate}} - 1 = \frac{16 \text{ MHz}}{4 * 115,2 \text{ kBd}} - 1 = 33,7\bar{2} \quad (4.13)$$

Unter Verwendung von 34 als nächstgelegenen ganzzahligen Wert für  $UxBRG$  ergibt sich die Baudrate:

$$\text{Baud Rate} = \frac{F_P}{4 * (UxBRG + 1)} = \frac{16 \text{ MHz}}{4 * (34 + 1)} = 114,285 \text{ kBd} \quad (4.14)$$

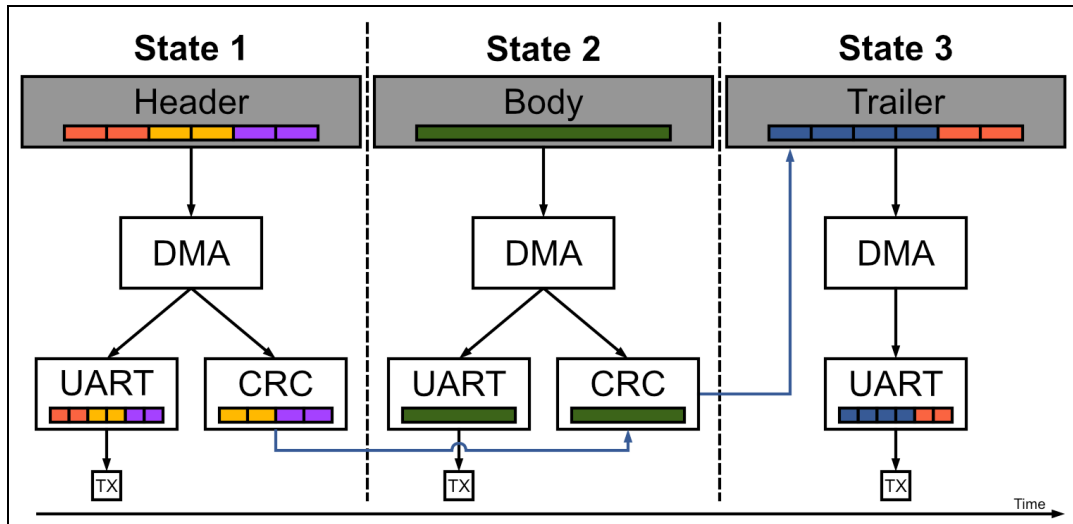
Der Vergleich zwischen der gewünschten Baudrate von 115,2 kBd und der in Gleichung 4.14 berechneten Baudrate ergibt den Fehler:

$$\text{Error} = 1 - \frac{114,285 \text{ kBd}}{115,2 \text{ kBd}} = 0,8\% \quad (4.15)$$

Microchip gibt keinen Toleranzbereich für die Abweichung der Baudrate des im Mikrocontroller integrierten UART-Moduls an. Daher wurde unter Verwendung von  $UxBRG = 34$  ein Funktionstest vorgenommen, der die fehlerfreie Übertragung bestätigt hat. Somit ist die Abweichung der Baudrate vom gewünschten Wert akzeptabel.

Die Abläufe zum Senden und Empfangen von Daten über die UART-Schnittstelle basieren auf dem in Abschnitt 4.1.6.1 beschriebenen Protokoll. Da die geringe Größe des SRAM keine Nutzung eines Sendepuffers erlaubt, ist der in Abbildung 4.5 dargestellte Frame hypothetisch in Header, Body und Trailer zu unterteilen.

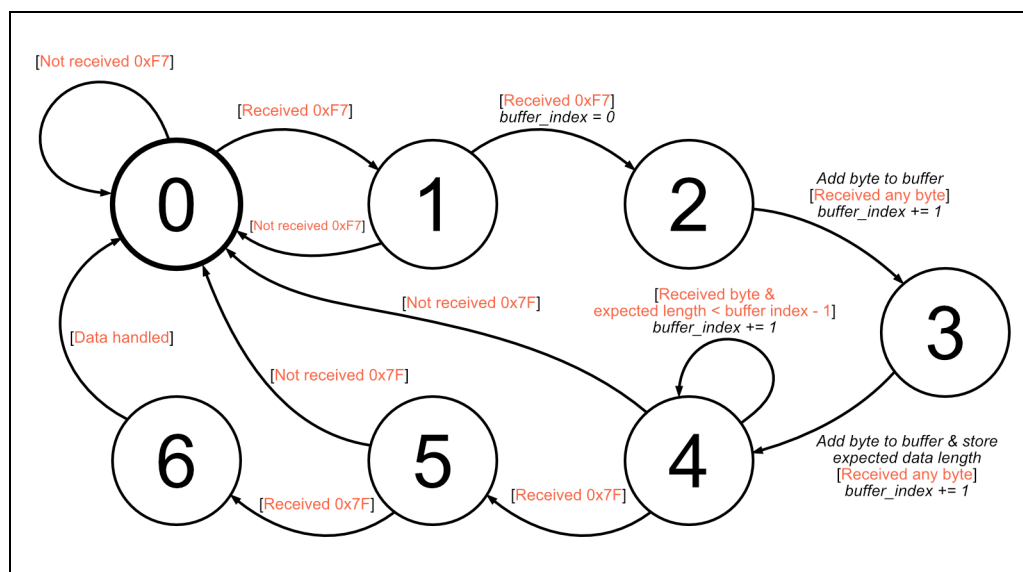
Während der Header das SOF-Symbol, die Datenlänge und den Datentyp umfasst, ist der Body der Payload gleichzusetzen. Somit ist dieser dynamisch einem beliebigen Speicherbereich im Mikrocontroller zuzuweisen und kein spezieller Puffer zu reservieren. Der Trailer schließt die CRC-Prüfsumme sowie das EOF-Symbol ein.



**Abb. 4.6:** Qualitativer Ablauf für das Senden eines Frames. Die farblichen Markierungen der einzelnen Daten entsprechen den in Abbildung 4.5 verwendeten. In den Prozess involviert sind die Direct Memory Access (DMA)-Peripherie, UART-Peripherie und CRC-Peripherie während TX den eigentlichen Datenbus kennzeichnet.

Abbildung 4.6 stellt den Ablauf zum Senden eines Frames unter Berücksichtigung der jeweiligen Zustände dar. Zur Minimierung der Systemauslastung werden die wesentlichen Prozesse zum Kopieren von Daten in die jeweiligen Peripherie-Register durch DMA-Peripherie ausgeführt. Diese kopiert die zu sendenden Bytes sukzessive in das Senderegister der UART-Peripherie sowie in das Schieberegister der CRC-Peripherie. Daten, welche nicht Teil des CRC sind, werden dementsprechend nur in das Senderegister der UART-Peripherie kopiert. Das Kopieren der Daten mittels DMA erfolgt ohne Nutzung von Ressourcen der CPU. Lediglich der Wechsel zwischen Header, Body und Trailer erfordert die erneute Konfiguration der DMA-Peripherie, welche innerhalb des Zustandsautomaten vorgenommen wird. Da neben dem SOF-Symbol der Trailer kein Bestandteil des CRC ist, wird vor dem Wechsel in den Zustand zum Versenden des Trailers die erzeugte CRC-Prüfsumme aus dem entsprechenden Register in den Trailer kopiert.

In der Implementierung ist durch dieses Verfahren die zu sendende Payload durch einen Zeiger zu übergeben, wodurch kein zusätzlicher Speicherplatz notwendig ist. Es bestehen lediglich Einschränkungen hinsichtlich der vorgegebenen Speicherausrichtung von 2 Byte sowie der maximalen Payload-Größe von 11 450 Byte. Entgegen der großen Datenmengen die gesendet werden können, ist die maximale Größe eines zu empfangenen Frames durch den Empfangspuffer auf 48 Byte beschränkt. Da die von der Visualisierungssoftware an das Gerät zu sendenden Datenpakete lediglich Einstellungen und Befehle enthalten, besteht allerdings keine Notwendigkeit für einen größeren Puffer.



**Abb. 4.7:** Zustandsdiagramm des Automaten zum Parsen von empfangenen Daten der UART-Schnittstelle. Orange Beschriftungen entsprechen Übergangsbedingungen. Kursive Beschriftungen entsprechen Aktionen.

Eingehende Daten werden von einem *Parser* verarbeitet und bei entsprechender Detektion eines Frames in den Puffer kopiert. Sobald der im Puffer gespeicherte Frame im Programmablauf verarbeitet wurde, wird der Parser zurückgesetzt, um neue Frames aus dem Datenstrom zu extrahieren. Aufgrund der Konfiguration der UART-Peripherie resultiert jedes empfangene Byte der UART-Schnittstelle in einem Aufruf der zugehörigen ISR, welche wiederum den Parser aufruft. Abbildung 4.7 zeigt die Funktionsweise des Parsers, welcher einem Zustandsautomaten entspricht.

Nach dem Warten auf das SOF-Symbol zu Beginn einer Nachricht wird aus der im Frame enthaltenen Information zur Länge der Daten die Position des EOF-Symbols



berechnet. Wird an Stelle eines erwarteten SOF-Symbols oder EOF-Symbols ein anderer Wert empfangen, setzt sich der Parser zurück und erwartet den Beginn einer neuen Nachricht. Da die CRC-Peripherie für das Senden von Daten verwendet wird, steht diese für das Empfangen von Daten nicht zur Verfügung. Deshalb wird zur Erhöhung der Effizienz auf eine softwareseitige Implementierung des CRC verzichtet und die Integrität der eingehenden Daten nicht überprüft. In Anbetracht der von der Visualisierungssoftware an das mobile NIR-Spektrometer zu übertragene Daten, welche lediglich Zustandsinformationen und Steuerbefehle enthalten (vgl. Tabelle 4.2), ist dies als akzeptabel anzuerkennen.

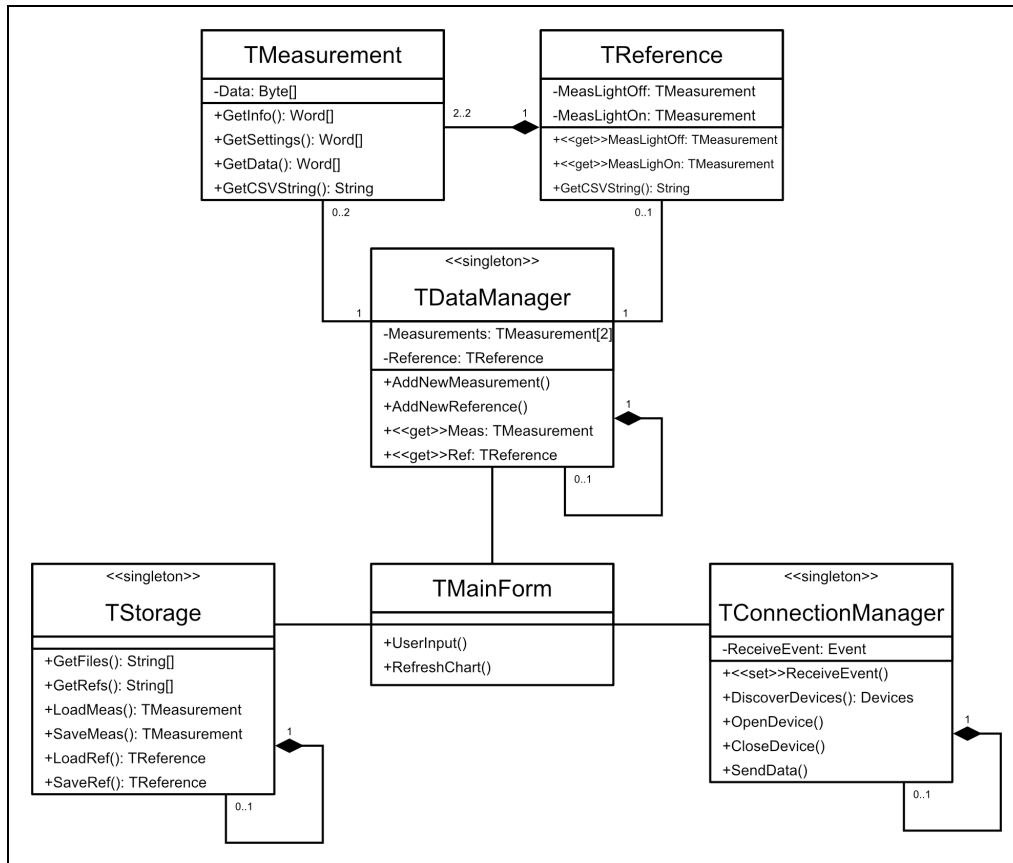
## 4.2 Visualisierungssoftware

Als Schnittstelle zwischen dem mobilen NIR-Spektrometer und dem Anwender ist die Visualisierungssoftware ein wesentlicher Bestandteil des Gesamtsystems. Neben dem Anzeigen und Verwalten von Messungen umfasst der Funktionsumfang auch die Steuerung des Spektrometers. Um die Mobilität bei aktiver Verbindung mit der Visualisierungssoftware zu erlauben, erfolgt die Implementierung der Software mit dem RAD Studio auf Basis des FireMonkey-Frameworks und Delphi, wodurch die plattformübergreifende Kompatibilität mit den Betriebssystemen Windows und Android gewährleistet ist (vgl. Abschnitt 2.2.2). In Kombination mit der Visualisierung von gespeicherten Messungen unterstützt die Software somit die Anwendung auf mobilen Computersystemen und Smartphones im Feld sowie die Nutzung auf stationären Plattformen zum Auswerten von bereits gesammelten Daten. Die detaillierten Funktionen und deren Umsetzung in der Software werden in den folgenden Abschnitten beschrieben und erläutert.

### 4.2.1 Programmstruktur

Der Programmablauf einer in Delphi programmierten Anwendung mit GUI ist ereignisbasiert und enthält daher keine Hauptschleife. Während die vom Benutzer getätigten Eingaben in der Anwendungsoberfläche entsprechende Funktionen zur Ereignisbehandlung aufrufen, ist für klassische Programmschleifen deren Umsetzung daher in Threads zu realisieren.

Die Struktur der objektorientierten Programmierung der Anwendung ist als Klassendiagramm darstellbar, welches in qualitativ-abstrahierter Form in Abbildung 4.8 aufgezeigt ist. Eine dem Quelltext entsprechende Darstellung der Klassen als Diagramm ist in Anlage K angefügt.



**Abb. 4.8:** Klassendiagramm der Visualisierungssoftware mit abstrahierten Klasseninhalten.

Neben den explizit als Singleton (zu Deutsch: Einzelexemplar) deklarierten Klassen TStorage, TDataManager und TConnectionManager ist durch die generelle Struktur einer Delphi-Anwendung festgelegt, dass in der Anwendung nur eine Instanz von TMainForm existieren darf. Somit unterscheiden sich diese Klassen von den als Datentyp verwendeten Klassen TMeasurement und TReference, deren Instanzen jeweils die Daten einer Messung beziehungsweise einer Referenzmessung enthalten und entsprechend mehrfach in der Laufzeit existieren.

Die Programmlogik ist vollständig in der Klasse TMainForm implementiert, da diese alle Komponenten der graphischen Oberfläche enthält und die Schnittstelle zur GUI ist. Während die Klasse TStorage alle Funktionen zum Laden und

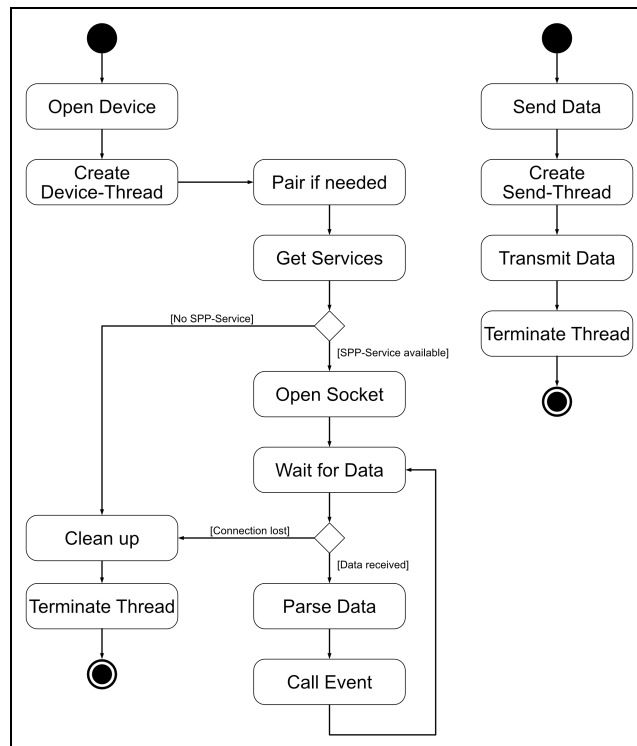
Speichern von Daten auf dem Dateisystem des Endgerätes inkludiert, organisiert die Klasse TDataManager die dem Benutzer aktuell angezeigten Messungen in der Visualisierung. Entsprechend der Bezeichnung sind alle Funktionen, die zur Bluetooth-Kommunikation mit dem Spektrometer notwendig sind, in der Klasse TConnectionManager zusammengefasst. Die Kommunikation zwischen den Klassen erfolgt sowohl durch Funktionsaufrufe mit Rückgabewerten, als auch durch Ereignisse denen bei der Instanziierung einer Klasse entsprechende Funktionen zur Ereignisbehandlung in TMainForm zugewiesen werden.

Ein wesentlicher Aspekt, der bei der plattformübergreifenden Entwicklung der Anwendung zu beachten ist, ist die Notwendigkeit der vom Anwender gewährten Berechtigungen zur Nutzung der Bluetooth-Peripherie und dem Zugriff auf das Dateisystem eines Gerätes mit Android-Betriebssystem. Da die Android-Anwendung keine Anforderung an die Lauffähigkeit im Hintergrund einer anderen Anwendung hat, startet diese nach einem Anwendungswechsel neu. Deshalb wird die Abfrage der entsprechenden Berechtigungen des Nutzers initial beim Start der Anwendung vorgenommen und dessen Ergebnis für die gesamte Laufzeit verwendet. Änderungen der Berechtigungen während der Laufzeit der Applikation und die daraus folgenden Einschränkungen der Funktionen sind demnach ausgeschlossen.

### **4.2.2 Datenübertragung**

Entgegen der in der Firmware implementierten UART-Schnittstelle, die mit dem Bluetooth-Modul kommuniziert (vgl. Abschnitt 4.1.6.2), ist in der Visualisierungssoftware die Bluetooth-Peripherie direkt anzusprechen. Die dazu notwendigen Funktionen werden vom FireMonkey-Framework über eine Systemklasse bereitgestellt und sind den Anforderungen entsprechend in der Klasse TConnectionManager implementiert. Neben dem Suchen nach sowohl dem Endgerät bekannten sowie unbekanntem Bluetooth-Geräten in der Umgebung, koordiniert die Klasse die Verbindung mit dem Spektrometer und das Senden und Empfangen von Daten nach dem im Abschnitt 4.1.6.1 beschriebenen Protokollformat. Bluetooth-Geräte werden in der Implementierung durch eine in der Systemklasse definierte Klasse dargestellt und als Parameter und Rückgabewerte der Klasse TConnectionManager sowie der entsprechenden Systemklasse verwendet.

Um durch die blockierenden Funktionen der Systemklasse beim Verbinden und Empfangen von Daten keine verzögernden Einflüsse auf die weiteren Programmabläufe zu erhalten, sind diese in einem Thread ausgelagert. Abbildung 4.9 zeigt das Aktivitätsdiagramm der zugehörigen Abläufe und zusätzlich den Ablauf zum Senden von Daten, der im Folgenden näher beschrieben wird.



**Abb. 4.9:** Aktivitätsdiagramm der Funktionen OpenDevice und SendData aus der Klasse TConnectionManager.

Da lediglich die Verbindung und Kommunikation zu einem Bluetooth-Gerät durch die Systemklasse erlaubt ist, darf nur ein Thread durch den Aufruf von OpenDevice aktiv sein. Ein Aufruf von OpenDevice bei bereits aktivem Thread zur Verbindung und Kommunikation mit einem Gerät beendet diesen und die damit verknüpfte Bluetooth-Verbindung. Anschließend wird ein neuer Thread mit dem entsprechenden Bluetooth-Gerät als Übergabeparameter erzeugt.

Dem Betriebssystem bekannte Geräte werden als paired (zu Deutsch: gepaart) bezeichnet und von der Systemklasse gesondert abgefragt. Ist das Bluetooth-Gerät dem Endgerät bisher nicht bekannt, wird die Verbindung erstmalig hergestellt und das Gerät im System des Endgerätes als bekannt (paired) vermerkt.

Zum Aufbau der seriellen Verbindung zwischen Spektrometer und Visualisierungssoftware muss das Bluetooth-Gerät das SPP unterstützen. Der entsprechende Service wird über eine Funktion der Systemklasse abgefragt und bei Verfügbarkeit über einen Socket (zu Deutsch: Kommunikationsendpunkt) gestartet. Anzumerken ist, dass der Verbindungsaufbau aufgrund der Auslegung des Kommunikationsprotokolls und des Verzichtes auf weitere Identifikationsverfahren mit jedem Bluetooth-Gerät mit SPP-Unterstützung möglich ist.

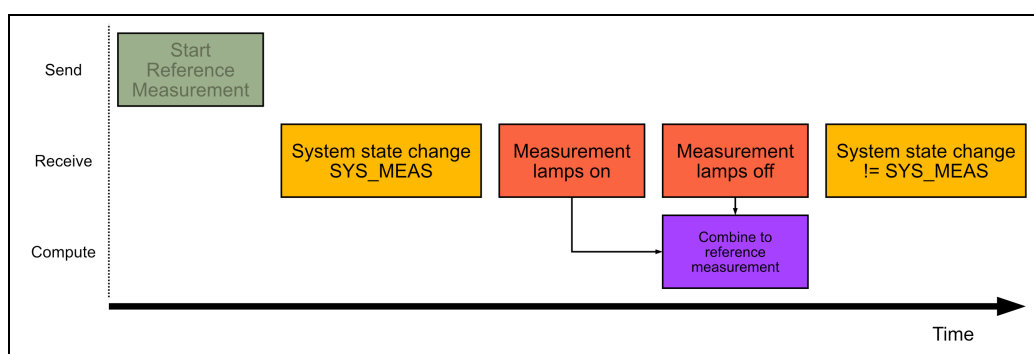
Sobald ein Fehler in der Verbindung oder Kommunikation mit dem Gerät auftritt, werden die verwendeten Ressourcen freigegeben und der Thread terminiert. Die übergeordnete Klasse TConnectionManager informiert daraufhin die Basisklasse TMainForm durch ein entsprechendes synchronisiertes Ereignis. Nach erfolgreichem Verbindungsaufbau zum Bluetooth-Modul des Spektrometers und Start des Sockets wird die blockierende Funktion der Systemklasse zum Empfangen von Daten aufgerufen. Da dieser Aufruf stets innerhalb des Threads erfolgt, bleiben neben dem Programmablauf der Basisklasse auch die weiteren Funktionen der Klasse TConnectionManager von dieser Blockierung unbeeinträchtigt. Während ein Verbindungsverlust zu der zuvor beschriebenen Terminierung des Threads führt, werden empfangene Daten im Parser verarbeitet.

Im Gegensatz zu dem in der Firmware des Spektrometers implementierten Parser, der lediglich sukzessiv empfangene Bytes verarbeitet (vgl. Abschnitt 4.1.6.2), muss der Parser der Visualisierungssoftware ein Array mit empfangenen Daten verarbeiten und zusätzlich den CRC ausführen. Unter Anwendung der größeren Speicherverfügbarkeit auf dem Endgerät der Visualisierungssoftware wird deshalb ein dynamischer Puffer in Form eines Arrays zum Zwischenspeichern von unverarbeiteten Daten verwendet. Dabei erweitern empfangende Daten den Puffer und erzeugen somit eine für den Parser zu verarbeitende Datenmenge, die aus mehreren empfangenen Frames beziehungsweise Framefragmenten bestehen kann.

Die grundlegende Detektion eines Frames durch den Parser folgt dem iterativen Prinzip des Zustandsautomaten des in der Firmware des Spektrometers angewandten Parsers und dessen struktureller Darstellung in Abbildung 4.7. Abweichend ist neben der Detektion des EOF-Symbols an dem erwarteten Index zusätzlich die Korrektheit des CRC zur Akzeptanz eines Frames notwendig. Der in Software

umgesetzte CRC verwendet entsprechend die in Abschnitt 4.1.6.1 beschriebenen Parameter. In dieser Arbeit stellt das Aufnehmen und Anzeigen gültiger Messwerte die Priorität dar, weshalb fehlerhafte Frames ignoriert und ohne weitere Behandlung verworfen werden. Daraus resultierende Fehler im Programmablauf, die aufgrund erwarteter, doch nicht eintreffender Nachrichten entstehen können, werden durch Starten oder Abbrechen einer Messung behoben. Dies begründet sich in der dadurch ausgelösten erneuten Übertragung des aktuellen Systemstatus des Spektrometers an die Visualisierungssoftware. Akzeptierte Frames werden der Klasse TMainForm über ein synchronisiertes Ereignis mitgeteilt.

Durch die Erläuterungen in Abschnitt 4.1.5 ist bekannt, dass eine Referenzmessung durch zwei einzeln übertragene Messungen mit verschiedenen Informationsparametern für den Status der Lampen übertragen wird. Die Betrachtung des Gesamtkontextes einer Referenzmessung beginnt mit dessen Auslösung durch eine Nachricht der Visualisierungssoftware oder die Betätigung des Tasters an dem Spektrometer. Da die Betätigung des Tasters am Spektrometer keine erneute Abfrage der aktuellen Konfiguration der Anzahl an Sample Points und Anzahl der Aufnahmen zur Mittelwertbildung von der Visualisierungssoftware anfordert, wird jede durch den Anwender ausgeführte Änderung dieser Parameter dem Spektrometer sofort mitgeteilt. Somit sind die eingestellten Werte unabhängig vom Auslöser einer Messung.



**Abb. 4.10:** Qualitative Übersicht des zeitlichen Ablaufs der Datenübertragung einer Referenzmessung. Dargestellt sind die ausgehend von der Visualisierungssoftware gesendeten und empfangenen Nachrichten sowie der Zeitpunkt der Erzeugung der Instanz von TReference (violett).

Der durch den Start einer Messung hervorgerufene Wechsel des Systemzustandes des Spektrometers wird als Nachricht von der Visualisierungssoftware empfangen

und löst entsprechende Aktionen zur visuellen Benachrichtigung des Anwenders aus. Wird die zweite Messung einer Referenzmessung empfangen, werden die darin befindlichen Informationen zum Status der Lampen jeder Messung abgeglichen und eine Instanz der Klasse TReference erzeugt. Entsprechend wird bei der Messung einer Probe eine Instanz der Klasse TMeasurement aus den empfangenen Daten erzeugt.

Das Senden von Daten über die Bluetooth-Schnittstelle erfolgt durch den Aufruf einer blockierenden Funktion der entsprechenden Systemklasse. Da die Delphi-RTL keine Implementierung einer threadsicheren Warteschlange zur Datensynchronisation mit einem Thread enthält, ist ein Verfahren zum nicht-blockierenden Senden von Daten festzulegen. Deshalb wird für jeden Aufruf zum Senden von Daten ein neuer Thread erzeugt. Dabei werden die zu sendenden Daten in einem Frame verpackt, die CRC-Prüfsumme berechnet und anschließend dem Thread bei seiner Instanziierung übergeben. Nach dem blockierenden Aufruf der entsprechenden Funktion zum Senden von Daten über die Bluetooth-Schnittstelle terminiert sich der Thread selbstständig. Somit ist stets gewährleistet, dass alle Aufrufe zum Senden von Daten verarbeitet werden und die aufrufende Funktion nicht blockieren muss. Unter Berücksichtigung der an das Spektrometer zu sendenden Daten, die im Wesentlichen Reaktionen auf Nutzereingaben darstellen und somit ein großes Sendeintervall haben, ist dieses Vorgehen als akzeptabel anzusehen.

### 4.2.3 Datenverwaltung

Neben der Visualisierung von eingelesenen Daten, welche detaillierter im folgenden Abschnitt 4.2.5 beschrieben wird, ist das Exportieren von Daten zur weiterführenden Analyse mit anderen Anwendungen eine Anforderung der Visualisierungssoftware. Wie bereits in Abschnitt 4.2.1 erwähnt, ist diese Anforderung erweitert ausgeführt und ermöglicht neben dem Exportieren beziehungsweise Speichern von Daten zusätzlich das Laden und erneute Visualisieren der geladenen Daten. Dazu werden Messungen und Referenzmessungen in Form von Comma-Separated Value (CSV)-Dateien auf dem Dateisystem des Endgerätes abgelegt. Dessen American Standard Code for Information Interchange (ASCII)-Kodierung erlaubt neben dem Einlesen in eine Vielzahl verschiedener Tabellenkalkulationsprogramme das

Einsehen der Daten mit einem Texteditor. Während zur Identifikation im Dateisystem die Dateiendung für Messungen „meas“ und die Dateiendung für Referenzmessungen „ref“ verwendet wird, beginnt der Inhalt einer gespeicherten Messung zur zusätzlichen Identifikation des Dateiinhaltes mit „\_BTAK;MEAS;20“. Der Inhalt einer gespeicherten Referenzmessung beginnt mit „\_BTAK;REF;20“. In der Darstellung als Tabelle werden Spalten durch Semikolon und Zeilen durch Zeilenumbrüche separiert, wodurch die genannten Identifikatoren den Inhalt von drei Spalten in der ersten Spalte einer Tabelle ergeben. Eine vollständige Beschreibung des Aufbaus der Dateien mit den zugehörigen Daten ist in Anlage L einzusehen.

Auf dem Dateisystem des Endgerätes der Visualisierungssoftware ist der Speicherort der Dateien unabhängig vom Betriebssystem das öffentliche Benutzerverzeichnis. In diesem erstellt die Anwendung ein Verzeichnis „NIR-Spectrometer-App“ mit entsprechenden Unterverzeichnissen „Measurement“ für Messungen und „Reference“ für Referenzmessungen. Da das Laden und Speichern von Messungen nur auf diese Verzeichnisse beschränkt ist, sind externe Dateien zur Visualisierung in der Anwendung darin abzulegen. Anzumerken ist, dass die Verzeichnisse durch den Zugriff auf das Dateisystem erstellt werden, welcher erstmalig durch Nutzereingaben zum Laden oder Speichern von Daten ausgeführt wird. Weitere Informationen zu den Funktionen der Benutzeroberfläche werden im folgenden Abschnitt erläutert.

Während sowohl die Erzeugung einer CSV-Zeichenkette, als auch die Extraktion der Daten aus einer CSV-Zeichenkette zum Speichern und Laden von Messungen und Referenzmessungen in den entsprechenden Klassen TMeasurement und TReference erfolgt, werden alle weiteren zuvor beschriebenen Umfänge, die Zugriff auf das Dateisystem tätigen, durch die Klasse TStorage zusammengefasst. Dazu zählen ebenfalls Nutzerabfragen zur Bestätigung des Überschreibens von Dateien bei äquivalenten Dateinamen. Basierend auf dem überladenen Konstruktor der Klassen TReference und TMeasurement erfolgt die weitere Handhabung der Daten in der Anwendung unabhängig davon, ob diese vom Dateisystem in Form einer CSV-Zeichenkette geladen oder direkt vom Spektrometer in einer Nachricht über die Bluetooth-Schnittstelle empfangen wurden. Bei der Instanziierung der Klasse TMeasurement werden dabei die in den Eingabedaten vorhandenen Informationen abhängig vom Format in die in Abschnitt 4.1 erarbeiteten Datenformate umgewan-



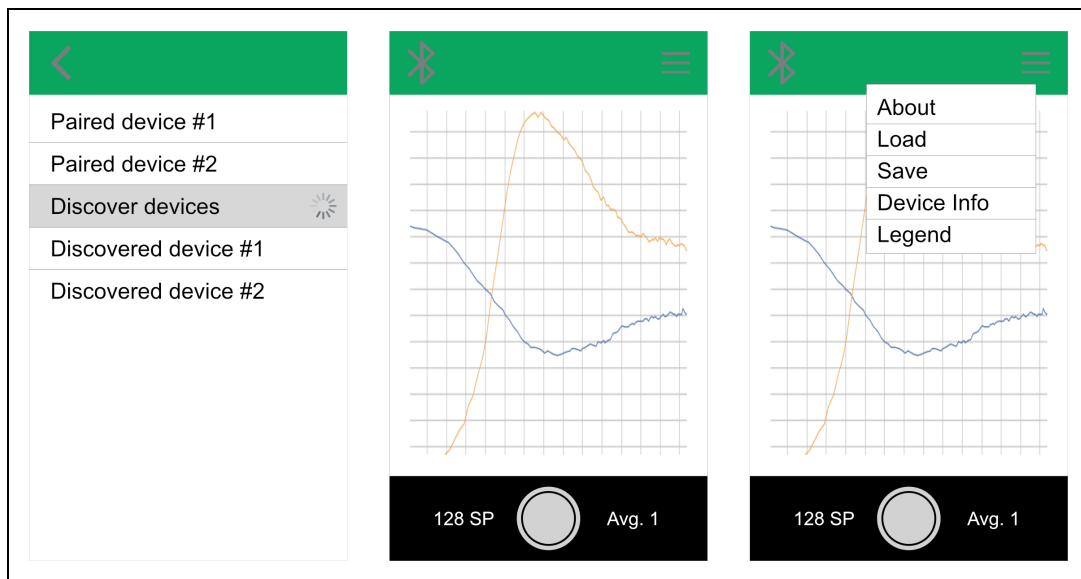
delt und innerhalb der Instanz gespeichert. Die Betrachtung der Attribute der Klasse TMeasurement in Anlage K zeigt die darin abgelegten Daten und deren Datenformate. Dabei sind die Datenformate der Klassenattribute identisch mit den in der CSV-Datei Verwendeten (vgl. Anlage L). Zu beachten ist, dass eine Instanz der Klasse TReference immer die Zeiger auf zwei Instanzen der Klasse TMeasurement enthält, in denen die Daten gespeichert sind und dessen Typumwandlung erfolgt (vgl. Abbildung 4.8).

#### 4.2.4 Benutzeroberfläche

Obwohl in dieser Arbeit die Funktionalität der Anwendung als Teil des Gesamtsystems die höchste Relevanz hat, ist für das Design der Benutzeroberfläche zusätzlich die intuitive Bedienbarkeit in Kombination mit einer attraktiven visuellen Darstellung priorisiert. Dabei ist eine Designlösung für die plattformübergreifende Kompatibilität unter Beibehaltung einer minimalen Komplexität der jeweiligen Darstellung zu entwickeln. Zu beachten sind die unterschiedlichen Verhalten der Formularelemente auf den verschiedenen Betriebssystemen, die trotz identischer Implementierung durch das FireMonkey-Framework bestehen. Als Beispiel ist die selektierte Zeile in einem Listenelement zu nennen, welche bei der Verwendung eines benutzerdefinierten Zeileninhalts in der Standardkonfiguration nur unter Windows visuell markiert wird. In diesem Fall ist die Implementierung zusätzlicher Funktionen für die äquivalente Darstellung in Android notwendig. Abbildung 4.11 zeigt die wesentlichen Ansichten der Anwendung als Designkonzept, welches gleichermaßen für Endgeräte mit dem Betriebssystem Windows und Android umgesetzt ist.

Zur intuitiven Bedienung der Steuerelemente zum Starten und Abbrechen einer Messung sowie dem Ändern der Parameter, ist dessen Design und Layout dem der von Smartphones bekannten Kamera-Applikationen nachempfunden. Während das Starten und Abbrechen einer Messung entsprechend dem an dem Spektrometer befindlichen Taster K1 (vgl. Abschnitt 4.1.2) durch die im Bedienfeld zentrale kreisförmige Schaltfläche ausgeführt wird, sind die in Abschnitt 4.1.6 beschriebenen einstellbaren Parameter einer Messung umliegend angeordnet. Nach dem Start der Anwendung sind diese auf „128 SP“ und „Avg. 1“ festgelegt. Dabei entspricht der

Wert vor „SP“ der Anzahl an Sample Points und der Wert nach „Avg.“ der Anzahl an Aufnahmen zur Mittelwertbildung. Wie bereits in Abschnitt 4.1.6 beschrieben, ist die Verzögerung zur Stabilisierung der Steuerspannung zwischen den Sample Points statisch implementiert und benötigt daher kein Eingabefeld in der Visualisierungssoftware. Die in der Anwendung zentrale Visualisierung der Messdaten in Liniendiagrammen sowie dessen weitere Zusammenhänge mit den einstellbaren Parametern werden im folgenden Abschnitt nähergehend erläutert.



**Abb. 4.11:** Designkonzept der Benutzeroberfläche. Ansicht zum Verbinden mit einem Spektrometer (Links). Visualisierung und Steuerung von Messungen im Hauptbildschirm (Mitte). Geöffnetes Menü für weitere Funktionen (Rechts).




Zu beachten ist, dass die Anwendung lediglich die Aufnahme einer einzelnen Messung durch das Auslösen über die Schaltfläche in der Visualisierungssoftware oder durch Betätigen des Tasters an dem Spektrometer erlaubt. Eine kontinuierliche Aufnahme an Messungen mit stetig aktualisierender Visualisierung ist aufgrund der durch die Parameter einstellbaren Messdauer von maximal etwa zwei Minuten nicht rentabel. Dargestellt wird dem Anwender eine aktive Messung in der Visualisierungssoftware durch die grüne Färbung der Schaltfläche zum Auslösen einer Messung.

Weitere Funktionen, die nicht direkt für die Aufnahme einer spektralen Verteilung relevant sind oder kein großes Nutzungsintervall aufweisen, sind über das Menü aufrufbar, welches über die rechts in der Navigationsleiste befindlichen Schalt-

fläche angezeigt wird. Dazu zählen alle Optionen der Datenverwaltung (vgl. Abschnitt 4.2.3), die Aufnahme einer Referenzmessung (vgl. Abschnitt 4.2.2) sowie das Anzeigen und Verbergen der Legende der Visualisierung. Weiterhin bietet der Menüpunkt „Device Info“ die Darstellung von Informationen zu dem aktuell verbundenen Spektrometer und dessen Status.

Neben dem Bluetooth-Gerätenamen und der Bluetooth-Adresse des Spektrometers zeigt die Ansicht zusätzlich dessen aktuellen Systemstatus und die Spannung der Batterie. Dabei werden diese Informationen sowohl von der Klasse `TConnectionManager`, als auch durch die verschiedenen Nachrichtentypen des Kommunikationsprotokolls (vgl. Abschnitt 4.1.6.1) bereitgestellt. Weiterhin ist über eine Schaltfläche das ferngesteuerte Ausschalten des Spektrometers möglich.

Jedes Ereignis in Form von Nutzereingaben oder vom Spektrometer empfangenen Informationen aktualisiert die Eingabemöglichkeiten. Somit ist zum Beispiel die zuvor beschriebene Ansicht ohne aktive Bluetooth-Verbindung mit einem Spektrometer nicht aufrufbar und die Auswahloptionen des entsprechenden Menüs beschränkt. Ebenfalls führt eine zu geringe Spannung der Batterie des Spektrometers zu einer Fehlermeldung, wodurch die Schaltfläche zum Starten von Messungen deaktiviert wird (vgl. Abschnitt 4.1.3). Dementsprechend stehen dem Anwender nur jene Funktionalitäten aktiv zur Verfügung, die unter Berücksichtigung des Zustandes des Gesamtsystems ausführbar sind.

Symbol	Information
	No adapter found
	No device connected
	Device connected

**Tab. 4.3:** Bedeutung der Bluetooth-Symbole in der Navigationsleiste.

Der Verbindungsstatus mit dem mobilen Spektrometer wird durch die entsprechenden Darstellungen des Bluetooth-Symbols auf der linken Seite der Navigationsleiste angezeigt. Tabelle 4.3 fasst die möglichen Symbole mit den entsprechenden Bedeu-

tungen zusammen. Neben der Anzeige des Verbindungsstatus dienen die Symbole gleichzeitig als Schaltfläche zum Verbinden oder Trennen eines Spektrometers. Während die Betätigung der Schaltfläche bei verbundenem Gerät die Verbindung nach der Bestätigung des Anwenders trennt, öffnet sich bei der Betätigung bei nicht verbundenem Gerät eine neue Ansicht. Die darin befindliche Liste zeigt alle Bluetooth-Geräte, die dem Endgerät bereits bekannt sind (vgl. Abschnitt 4.2.2) und erlaubt über die integrierte Schaltfläche „Discover devices“ die Suche nach weiteren in der Umgebung befindlichen Geräten. Durch Auswahl eines Listeneintrages wird die Verbindung zu dem entsprechenden Gerät hergestellt.

### 4.2.5 Visualisierung der Daten

Die Verwaltung der in der Anwendung visualisierten Messungen erfolgt durch die Klasse TDataManager, welche Zeiger auf die Instanzen von maximal einer Referenzmessung und maximal zwei Messungen von Proben enthält. In der Implementierung werden die Zeiger auf die Instanzen der Messungen von Proben in einem Ringpuffer abgelegt, wodurch das Hinzufügen einer neuen Messung zu dem rotierenden Überschreiben der vorhandenen Messungen führt. Zum Erhalt der Übersichtlichkeit der Anwendung auf mobilen Endgeräten und entsprechend der in Abschnitt 3.3 beschriebenen Anforderung ist die Puffergröße auf zwei Instanzen beschränkt.

Alle gleichzeitig in der Visualisierung dargestellten Messungen sowie die Referenzmessung müssen die identische Anzahl an Sample Points aufweisen. Bei versuchter Änderung der entsprechenden Einstellung oder durch Laden einer Messung mit abweichender Anzahl an Sample Points muss der Anwender dieser zustimmen, wodurch alle Daten aus der Visualisierung entfernt werden. Durch diese Bedingung entfällt die Gefahr, dass Messungen mit abweichender Anzahl an Sample Points aufgrund der möglichen Datenlücken im visuellen Vergleich falsch interpretiert werden. Eine Beschränkung für die Äquivalenz der Anzahl an Aufnahmen zur Mittelwertbildung besteht für die visualisierten Messungen nicht.

Wie bereits in Abschnitt 4.2.4 erwähnt, erfolgt die letztendliche Visualisierung der Messdaten in Form von Liniendiagrammen, welche dem Anwender zur individuellen Betrachtung der Messungen vier verschiedene Ansichtsmodi bieten. Neben

der Darstellung der sogenannten Rohdaten erlaubt die Anwendung die Darstellung der spektralen Verteilung als Transmission, Extinktion sowie der Extinktion mit angeglichenener Nulllinie zum einfachen Vergleich zweier Messungen.

Durch Auswahl des Reiters „Raw“ ist die Darstellung der durch die Klasse TDataManager verwalteten Messungen in einem Liniendiagramm möglich. Dabei ist der digitale Wert des SD-ADC unter Berücksichtigung der in Abschnitt 4.1.4.2 durch die mögliche Mittelwertbildung mehrerer Aufnahmen einer Messung erläuterten und verwendeten Festkommadarstellung gegenüber dem Index der Sample Points aufgetragen. Zum einen erlaubt diese Ansicht der Rohdaten dem Anwender die visuelle Validierung der Referenzmessung, zum anderen lassen sich diese Werte zur weiteren Optimierung der Hardware nutzen. Basierend auf der formalen Definition der Extinktion (vgl. Abschnitt 3.3) muss der Signalpegel der Messung einer Referenzmessung mit eingeschalteter Beleuchtung über den gesamten Messbereich größer als der Signalpegel der Messungen von Proben sein. Gleichmaßen muss der Signalpegel der Messung der Referenzmessung mit ausgeschalteter Beleuchtung für gültige Ergebnisse stets kleiner sein als der Signalpegel von Messungen von Proben. Anzumerken ist, dass diese Bedingungen weder von der Anwendung geprüft werden, noch der Anwender auf diese hingewiesen wird.

Sobald eine Referenzmessung entweder durch das Laden vom Dateisystem des Endgerätes oder durch das Erstellen über den entsprechenden Menüpunkt in der Anwendung visualisiert ist (vgl. Abschnitt 4.2.4) und mindestens eine weitere Messung einer Probe geladen oder aufgenommen wurde, erlaubt die Anwendung die Auswahl des Reiters „Spectrum“. Durch dessen Auswahl wechselt die Ansicht zur weiteren Darstellung eines Liniendiagramms mit Auftragung der Wellenlänge auf der horizontalen Diagrammachse und den drei Darstellungsmöglichkeiten „ $I / I_0$ “, „ $\lg(I_0 / I)$ “ und „ $\Delta$  of  $\lg(I_0 / I)$ “ für die Werte der vertikalen Diagrammachse. Die Beschriftung der Reiter mit der formalen Darstellung der entsprechenden Darstellungsform begründet sich durch den geringen verfügbaren Platz in der Darstellung auf einem mobilen Endgerät wie einem Smartphone. Gleichmaßen entfällt aus diesem Grund die Beschriftung der vertikalen Diagrammachse.

Durch Auswahl des Reiters „ $I / I_0$ “ erfolgt entsprechend der formalen Bedeutung die Darstellung der Messungen als spektrale Verteilung durch Auftragung der

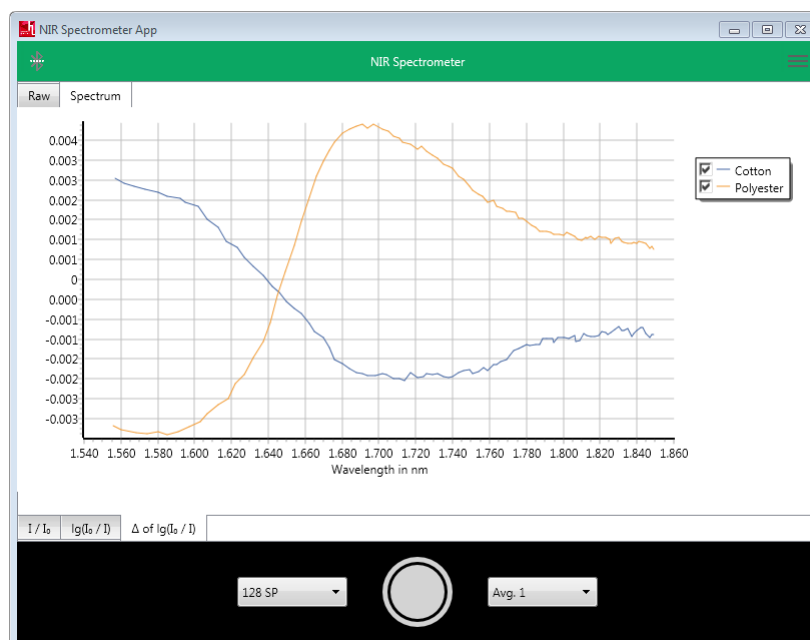
Transmission gegenüber der Wellenlänge (vgl. Abschnitt 2.1.1). Die Berechnung der Transmission erfolgt dabei durch Anwendung der Beziehung zwischen Transmission und Extinktion auf Basis der in Abschnitt 3.3 beschriebenen Definitionen. Somit begründet sich zudem die Notwendigkeit einer Referenzmessung zum Wechsel in die Ansicht „Spectrum“. Aufbauend darauf ändert sich der Inhalt des Liniendiagramms durch Auswahl des Reiters „ $\lg(I_0 / I)$ “ und stellt die Messungen als spektrale Verteilung durch Auftragung der Extinktion gegenüber der Wellenlänge dar (vgl. Abschnitt 2.1.1).

Aufgrund der möglichen Differenzen der Signalpegel von gleichzeitig dargestellten Messungen von verschiedenen Proben ist unter Umständen keine parallele und detaillierte Betrachtung der Messdaten möglich. Deshalb ist durch Auswahl des Reiters „ $\Delta$  of  $\lg(I_0 / I)$ “ die Darstellung der spektralen Verteilung als Extinktion mit angepasster Nulllinie der Messungen möglich. Dabei wird basierend auf Maximalwert und Minimalwert der Messungen jeweils deren arithmetisches Mittel bestimmt und die von diesem Wert ausgehende Abweichung von jedem Sample Point ausgehend von der Nulllinie aufgetragen. Zu berücksichtigen ist, dass diese Darstellung keiner formalen Grundlage folgt, sondern lediglich eine Möglichkeit zur visuellen Betrachtung von Kurvenverläufen in verschiedenen Wertebereichen von Signalpegeln ist.

Abbildung 4.12 zeigt die Visualisierungssoftware auf einem Endgerät mit Windows-Betriebssystem im Ansichtsmodus „Spectrum“ mit den beispielhaften Messergebnissen von Baumwolle und Polyester. Weitergehend sind die Bedienelemente zum Wechsel der möglichen Ansichten der Messdaten sowie die Legende sichtbar, welche das Einblenden und Ausblenden einzelner Kurven des Liniendiagramms erlaubt. Zur besseren Übersicht auf Endgeräten mit Android-Betriebssystem werden durch die Rotation des Gerätes in das Querformat alle Bedienelemente ausgeblendet.

Unabhängig von der gewählten Ansicht stehen dem Anwender weiterhin alle Funktionen zum Verwalten sowie Erstellen von Messungen zur Verfügung. Jede Änderung der Messdaten führt zur einer unverzüglichen Aktualisierung der graphischen Darstellung, welche durch die TChart-Bibliothek bereitgestellt wird. In der Implementierung erlaubt diese die einfache Konfiguration des gewünschten

Diagrammtyps und dessen Eigenschaften sowie Funktionen, wie die automatische Skalierung der Diagrammachsen. Sobald die Programmlogik in der Klasse TMainForm die Aktualisierung der Diagrammdaten erfordert, werden die verfügbaren Daten aller Messungen inklusive der Referenzmessung von der Klasse TDataManager abgefragt und für die Visualisierung vorbereitet. Um sowohl die Darstellung der Rohdaten, als auch die Darstellung der Daten als spektrale Verteilung stets synchron zu halten, wird bei einer Aktualisierung der Daten auch das dem Anwender nicht sichtbare Diagramm aktualisiert. Basierend auf der Leistungsfähigkeit der Endgeräte und der geringen Menge an zu verarbeitenden Daten entsteht dem Anwender dadurch kein wahrnehmbarer Nachteil.



**Abb. 4.12:** Darstellung der aufgenommenen spektralen Verteilungen von Baumwolle (blau) und Polyester (orange) in der Vergleichsansicht der Visualisierungssoftware auf einem Endgerät mit Windows-Betriebssystem.

Aufgrund der Notwendigkeit einer Referenzmessung zur Berechnung der Wellenlänge aus der gemessenen Steuerspannung des FPI im NIR MEMS-FPI Spektroskop zur Darstellung der Ergebnisse als spektrale Verteilung (vgl. Abschnitt 3.2.4) erfolgt diese ebenfalls bei jeder Aktualisierung von Messungen. Da alle notwendigen Informationen durch die von der Klasse TDataManager bereitgestellten Zeiger auf die Instanzen der Klassen TMeasurement und TReference verfügbar sind, werden lediglich die Berechnung der Wellenlänge sowie die Berechnung der

---

entsprechenden durch den Anwender ausgewählten Darstellungsform sukzessive für jeden Sample Point ausgeführt und anschließend der TChart-Komponente zur Visualisierung übergeben.

Abschließend ist zu beachten, dass sowohl die Daten der Steuerspannung des FPI, als auch die Temperatur des Sensors während einer Messung nicht direkt in der Anwendung visualisiert werden. Allerdings sind diese, wie bereits in Abschnitt 4.2.3 beschrieben, durch Speichern der entsprechenden Messung auf dem Dateisystem des jeweiligen Endgerätes einzusehen.



## 5 Evaluation des Systems

Basierend auf der abgeschlossenen Implementierung von sowohl der Firmware als auch der Visualisierungssoftware ist das Gesamtsystem zu evaluieren. Dazu sind neben der Kontrolle der Funktionsfähigkeit der beiden Softwareapplikationen die durch die Messung von Proben ermittelten Daten visuell zu betrachten.

In Bezug auf die Softwareumfänge wurden alle in den vorherigen Kapiteln dieser Arbeit beschriebenen Funktionen und Funktionalitäten umgesetzt und im Rahmen der Evaluation geprüft. Während die korrekte Funktionsweise einzelner Komponenten von sowohl Firmware als auch Visualisierungssoftware bereits während der Implementierung erfolgte, wurde die Funktionsprüfung des Gesamtsystems in der Anwendung beim Sammeln der Messergebnisse für die Evaluation durchgeführt. Aufgrund der Einbettung der Firmware in das System des Spektrometers und der daraus resultierenden engen Kommunikation zwischen Hardware und Software umfasst das Ergebnis der Prüfung ebenfalls die Funktionalität der hardwareseitigen Umfänge. Für die in Abschnitt 4.1.4 beschriebene Verzögerung zwischen den Sample Points wurde ermittelt, dass der durch das Intervall des SD-ADC vorgegebene Zeitverzug von 2,56 ms zur Vorbereitung und Verarbeitung jedes Sample Points für die Stabilisation der Steuerspannung des FPI ausreichend ist. Demnach ist der entsprechende Parameter in der Firmware mit dem Wert 0 belegt. Weitergehend hat sich für die Verzögerung nach der Messung mit eingeschalteten Lampen bei einer Referenzmessung ein Wert von 250 (entspricht 320 ms) als akzeptabel erwiesen.

Im Verlauf der Funktionsprüfung hat sich für die Visualisierungssoftware in Kombination mit einem Android-Betriebssystem ergeben, dass die Anwendung nach durchschnittlich neun Minuten abstürzt. Während dieses Verhalten unter Verwendung von Endgeräten mit Windows-Betriebssystem nicht aufgetreten ist, konnte der Fehler, welcher im Debugger als Stack-Überlauf deklariert wurde, nur bei aktiver Bluetooth-Verbindung reproduziert werden. Da der Debugger keine weiteren Informationen über die Ursache oder Position im Quelltext ausgegeben hat, wurde das Programm auf unbekannte rekursive Funktionsaufrufe basierend auf der

Bluetooth-Verbindung untersucht. Nachdem keine rekursiven Funktionsaufrufe gefunden wurden, hat sich gezeigt, dass der Absturz nicht reproduzierbar ist, sobald die Ansicht zum Verbinden von Bluetooth-Geräten nach dessen Verbindung nicht geschlossen wird. Der ursprüngliche Programmablauf sieht vor, nach erfolgreicher Verbindungsherstellung automatisch in die Ansicht zur Visualisierung der Messdaten zu wechseln. Die genaue Ursache des Programmabsturzes konnte auch nach weiteren Untersuchungen nicht identifiziert werden, allerdings wurde der Fehler durch die Deaktivierung des automatischen Wechsels der Ansichten unterbunden. Für den Anwender ergibt sich lediglich die Notwendigkeit, nach der Verbindungsherstellung die Ansicht manuell zu wechseln.

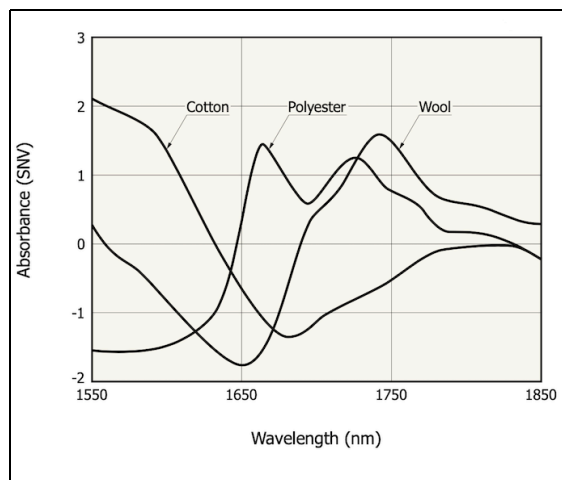
In Abschnitt 3.2.2 wurde bereits beschrieben, dass spektrale Verteilungen, die mit unterschiedlichen Lampen aufgenommen wurden, nicht direkt miteinander verglichen werden können. Dies ist durch die unterschiedlichen Einflüsse des Messsystems begründet, welche unter anderem durch die spektrale Verteilung des ausgestrahlten Lichts der Lampen entstehen. Da das Ziel dieser Arbeit keine Analyse der Messdaten beinhaltet, ist für den Vergleich von spektralen Verteilungen ausschließlich eine qualitative und keine quantitative Bewertung zu treffen. Diese ist visuell zu erfassen und vergleichsweise zu interpretieren.

Um eine minimale Menge an spezifischen Einflüssen des Messsystems zu erhalten, bietet sich der Messdatenvergleich mit einem äquivalenten und bereits validierten System unter Verwendung des identischen Sensortyps an. Die Veröffentlichung der proprietären Lösung des mobilen NIR-Spektrometers von Hamamatsu im Verlauf dieser Arbeit ermöglichte den Zugriff auf dessen im Technical Note (vgl. [19]) ergänzten Beispielmessungen von Textilien und Polymeren. Basierend auf der Anwendung des identischen Sensortyps zur Aufnahme der Beispielmessungen im Technical Note sind die darin dargestellten Diagramme als Vergleich zu nutzen. Entgegen der in der Visualisierungssoftware möglichen Darstellung spektraler Verteilungen als Extinktion, werden diese in dem Technical Note als Extinktion mit Standard Normal Variates (SNV)-Transformation dargestellt. Dabei wird die SNV-Transformation zur Korrektur von Effekten wie der Verschiebung der Basislinie durch unspezifische Lichtstreuung an Probenpartikeln angewandt [16, S. 161]. Die Betrachtung der formalen Definition der SNV Transformation in Gleichung 5.1 zeigt, dass diese keinen Einfluss auf den Kurvenverlauf und somit die Vergleich-

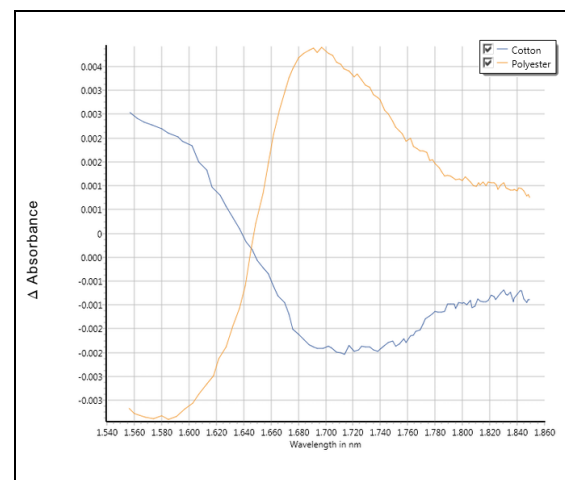
barkeit mit der Darstellung einer spektralen Verteilung in Extinktion hat:

$$x_{i,j}^{SNV} = \frac{(x_{i,j} - \bar{x}_i)}{\sqrt{\frac{\sum_{j=1}^p (x_{i,j} - \bar{x}_i)^2}{p-1}}} \quad (5.1)$$

Übertragen auf die Anwendung einer Messreihe in Extinktion entspricht  $x_{i,j}^{SNV}$  der transformierten Extinktion eines Sample Points. Neben dem originalen Signalwert eines Sample Points  $x_{i,j}$  beschreibt  $\bar{x}_i$  das arithmetische Mittel der spektralen Verteilung und  $p$  die Anzahl an Sample Points. Als Resultat der SNV-Transformation ergibt sich eine Verschiebung der Basislinie sowie die Änderung der Signalamplitude und die Minimierung von Signalrauschen. Für den visuellen Vergleich des Kurvenverlaufes sind diese Informationen allerdings nicht relevant, wodurch dieser ohne die weitere Bearbeitung der Daten möglich ist. Um die Qualität der Messdaten bei der zukünftigen Weiterentwicklung der Visualisierungssoftware zu verbessern, ist als Optimierungspotenzial die Anwendung der SNV-Transformation zu erwägen.



**Abb. 5.1:** Spektrale Verteilung von Baumwolle, Polyester und Wolle [19].



**Abb. 5.2:** Auszug aus der Visualisierungssoftware mit Darstellung der spektralen Verteilungen von Baumwolle und Polyester.

Abbildung 5.1 zeigt die visuelle Darstellung der Messdaten von Baumwolle, Wolle und Polyester aus dem Technical Note des Hamamatsu Sensors (vgl. [19]) ohne weitere Angabe der Aufnahmeparameter. Gleichmaßen sind in Abbildung 5.2 die Messdaten von Baumwolle und Polyester dargestellt, welche mit dem mobilen

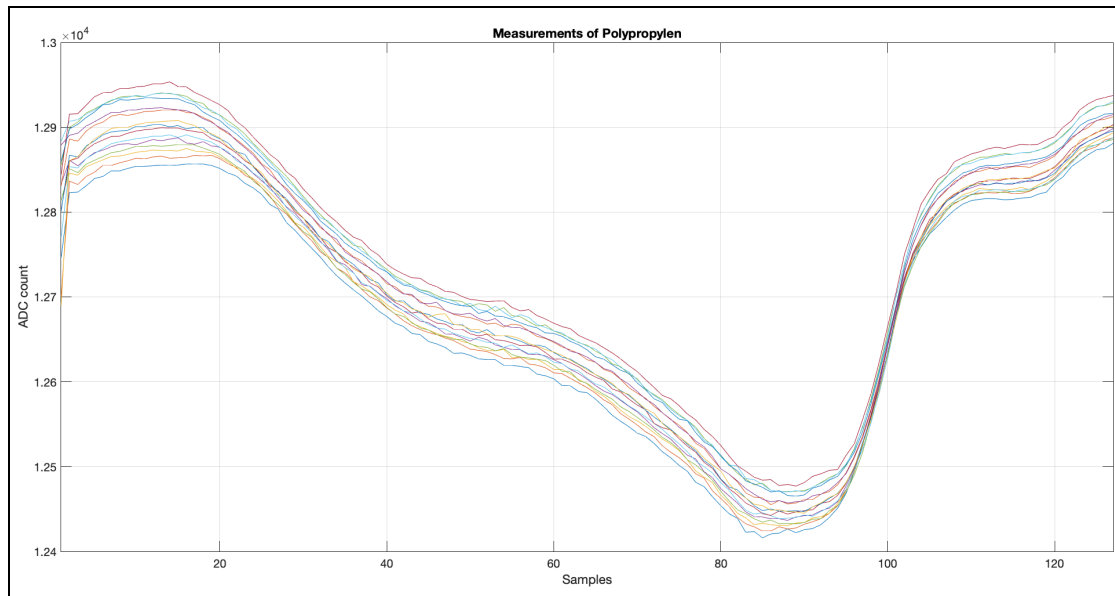
NIR-Spektrometer unter Verwendung der in dieser Arbeit entwickelten Software aufgenommen und visualisiert wurden. Dabei wurden die Messungen mit 128 Sample Points bei einfacher Aufnahme einer Messung mit der Referenzmessung von Aluminiumoxid vorgenommen. Unter der Vernachlässigung des Rauschens, welches aufgrund des Vergleiches zwischen Extinktion mit SNV-Transformation und Extinktion nicht betrachtet werden kann, zeigt die visuelle Betrachtung der jeweiligen Messungen der zwei Abbildungen Übereinstimmungen der Kurvenverläufe. Weiterhin ist die visuelle Differenzierung der Proben durch die Messdaten bei der entwickelten Lösung ebenso erwiesen, wie bei der proprietären Lösung.

Markant ist allerdings der geringe Detailanteil und der daraus resultierende Informationsverlust der Messungen, die in dieser Arbeit entwickelten Software dargestellt werden. So ist bei der Betrachtung des Kurvenverlaufes ohne Beachtung der Wellenlänge die Messung von Polyester mit der entwickelten Lösung nicht von der Messung von Wolle mit der proprietären Lösung differenzierbar. Entsprechend der Angabe des Arbeitsbereiches des verwendeten Sensors (vgl. Abschnitt 2.2.1) sind die Erkenntnisse der Messungen von Textilien äquivalent auf Messungen für Polymere zu übertragen.

Während Hamamatsu in dem Technical Note der Baureihe des Sensor einen Lichteinfallswinkel von  $0^\circ$  und somit den senkrechten Lichteinfall empfiehlt (vgl. [19]), erlaubt die Anordnung der Sensorik des mobilen NIR-Spektrometers ein breites Spektrum an Lichteinfallswinkeln. Basierend auf dem Verlust von Details bei der Messung spektraler Verteilung, ist von dem Einfluss des Lichteinfallswinkels auf die spektrale Auflösung auszugehen, der ebenfalls in dem Technical Note publiziert wird. Da die hardwareseitige Optimierung des Spektrometers keinen Umfang dieser Arbeit darstellt, ist das Anbringen einer Blende vor dem Sensor zur Einschränkung des Lichteinfallswinkels als Optimierungspotenzial für zukünftige Arbeiten an dem Projekt zu nennen.

Weitergehend ist neben der Betrachtung einzelner Messungen dessen Reproduzierbarkeit als Maß für die Stabilität des Systems zu prüfen. Dabei wird parallel zu dem Kommunikationsverhalten zwischen Spektrometer und Visualisierungssoftware der visuelle Vergleich mehrerer, aufeinanderfolgender Messungen einer Probe vorgenommen, wobei diese jeweils neu auf der Sensorik positioniert wird. Um die

in Abschnitt 3.2.3 beschriebene Erwärmung der Sensorik als negativen Einfluss auf die Reproduzierbarkeit zu inkludieren, wurde die Messreihe innerhalb von 15 Minuten direkt nach dem Verbinden der Visualisierungssoftware mit dem Spektrometer vorgenommen. Unter Verwendung eines Messintervalls von einer Minute bei 128 Sample Points und der einfachen Aufnahme einer Messung ergeben sich die in Abbildung 5.3 dargestellten Rohwerte als Ergebnisse der 15 Messungen.

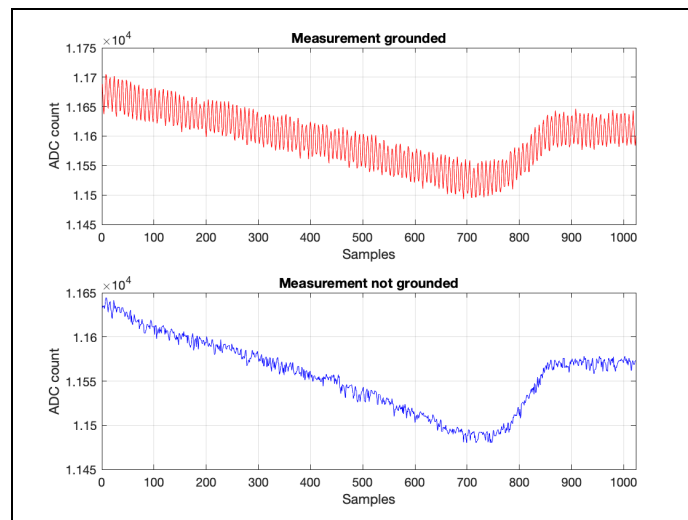


**Abb. 5.3:** Rohdaten von 15 Messungen von Polypropylen (PP) mit einem Messintervall von einer Minute.

Entsprechend der Stabilität der Datenübertragung und des Gesamtsystems, die durch die fehlerfreie Kommunikation zwischen Spektrometer und Visualisierungssoftware während der Messreihe bestätigt wird, zeigt die visuelle Betrachtung der Messergebnisse äquivalente Kurvenverläufe. Obwohl die einzelnen Messungen Versätze der Signalintensität sowie Änderungen der Amplituden aufweisen, sind die mit dem Spektrometer aufgenommenen Messungen mit einer akzeptablen Reproduzierbarkeit zu bewerten. Die spezifischen Analysen entsprechender Kenngrößen und mathematische Optimierungsverfahren bilden keinen Umfang dieser Arbeit und bleiben somit der zukünftigen Weiterentwicklung des Gesamtsystems vorbehalten.

Während bei der Nutzung des Spektrometers im Batteriebetrieb keine äußeren Einflüsse auf das Messsignal festgestellt werden konnten, führt die Verbindung mit einem geerdeten Gerät über dessen USB-Schnittstelle zu starkem Signalrauschen.

Unter der Berücksichtigung, dass durch die USB-Verbindung des Spektrometers mit einem nicht geerdeten Gerät keine Auswirkungen auf das Messsignal entstehen, ist von dem negativen Einfluss der Netzfrequenz von 50 Hz auszugehen. Abbildung 5.4 zeigt den Vergleich der Messdaten von zwei Messungen, welche den Effekt basierend auf der Erdung des Spektrometers darstellen. Der Ausschluss der softwareseitigen Einbringung eines Störsignals hat in der darauf folgenden Analyse der Hardware die Störanfälligkeit der Leiterbahn des Sensorsignals ergeben. Da durch den möglichen Batteriebetrieb keine Einschränkung hinsichtlich der Funktionalität des Gesamtsystems besteht und die Optimierung der Leiterplatte in dieser Arbeit nicht behandelt wird, ist dieses Verhalten im Umfang dieser Arbeit lediglich als bekannt zu vermerken.



**Abb. 5.4:** Rohdaten der Messungen einer undefinierten Probe mit und ohne Erdung des Spektrometers.

## 6 Fazit

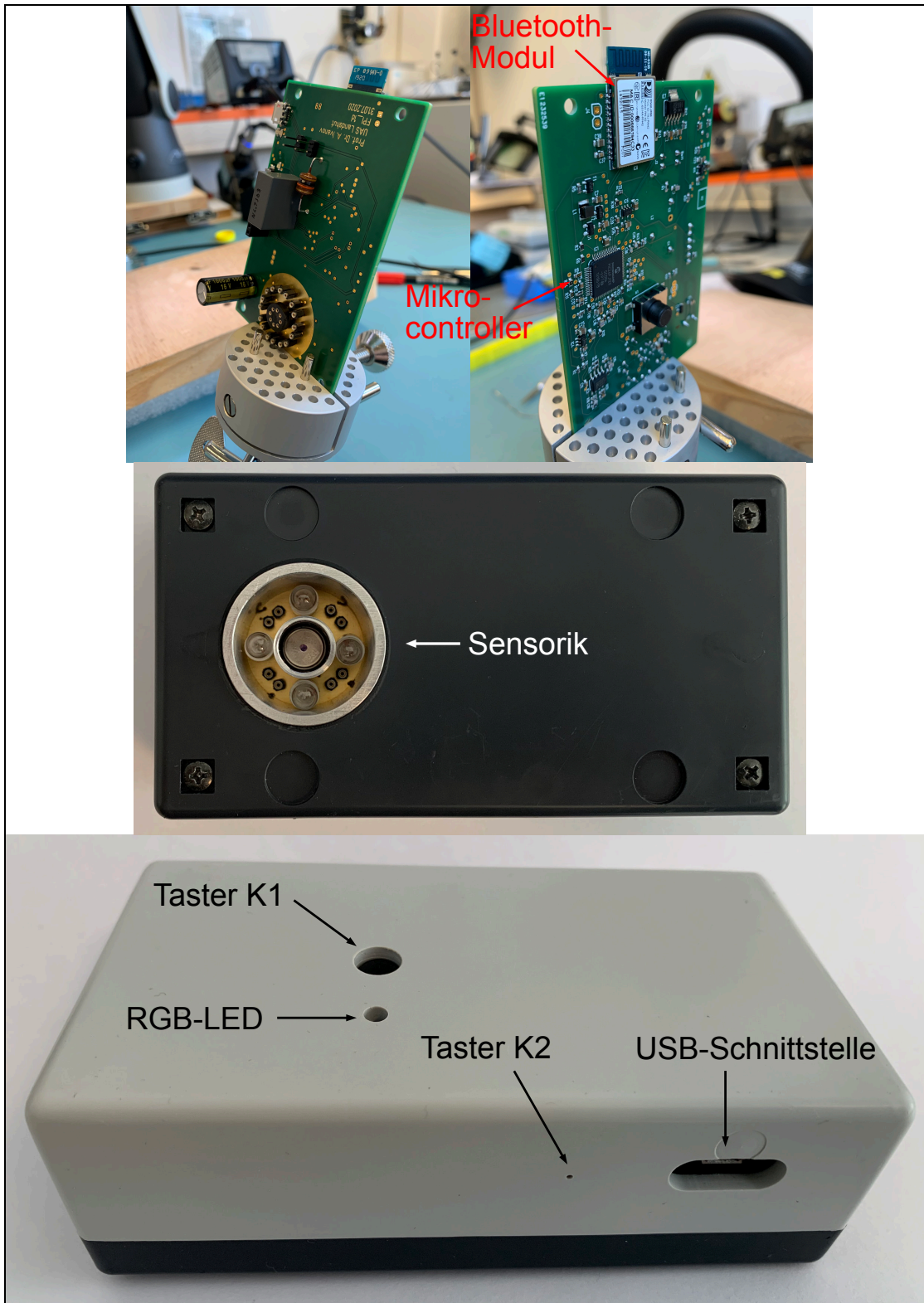
Das Ziel dieser Arbeit war die Bereitstellung von Softwarelösungen für die Firmware und Visualisierungssoftware eines mobilen NIR-Spektrometers, um dessen Fähigkeit im Kontext des Gesamtsystems zu bestimmen. Dabei wurden unter gegebener Hardware alle relevanten Informationen zu Eigenschaften und der Funktionsweise der NIR-Spektroskopie sowie der Applikation des verwendeten Sensors ermittelt und daraus ein Systemkonzept entwickelt. Neben dem Einlesen der Sensordaten und dessen Aufbereitung für die visuelle Darstellung in der jeweiligen Softwarekomponente war die Kommunikation zwischen Firmware und Visualisierungssoftware ein wesentlicher Faktor für die fehlerfreie Funktion des Gesamtsystems. Dazu war die Entwicklung eines Protokolls zur Übertragung verschiedener Informationen in Form von Messdaten und Steuerbefehlen zwischen den Geräten erforderlich. Die grundsätzliche Anforderung, spektrale Verteilungen von aufgenommenen Daten visuell darzustellen, wurde durch erweiterte Funktionalitäten der Datenverwaltung zum Laden und Speichern von Messungen in der Visualisierungssoftware ergänzt. In der abschließenden Evaluation wurden das Resultat der entwickelten Software sowie die Fähigkeit des Gesamtsystems beschrieben und Optimierungspotenziale aufgezeigt.

Durch die Ergebnisse der Evaluation und die erfolgreiche Umsetzung der Firmware und Visualisierungssoftware wurde bestätigt, dass die Fähigkeit des Gesamtsystems die Differenzierung verschiedener Kunststoffe und Textilien erlaubt. Während der in der Evaluation erkannte Softwarefehler behoben werden konnte, ist die Optimierung der hardwareseitigen Einflüsse nicht Teil des Umfangs dieser Arbeit und daher Inhalt für die zukünftige Weiterentwicklung des Systems. Basierend auf den möglichen Optimierungen ist mit der Applikation der modernen NIR MEMS-FPI Sensortechnologie unter Verwendung einfacher Komponentenstrukturen das Potenzial für die Materialidentifikation gegeben. Unter Betrachtung aller genannten Aspekte ist das Ziel dieser Arbeit vollumfänglich erreicht worden und sowohl die Informationsbasis als auch die Hardwarebasis für die weitergehende Entwicklung des Gesamtsystems mit dem Ziel der Materialidentifikation gegeben.

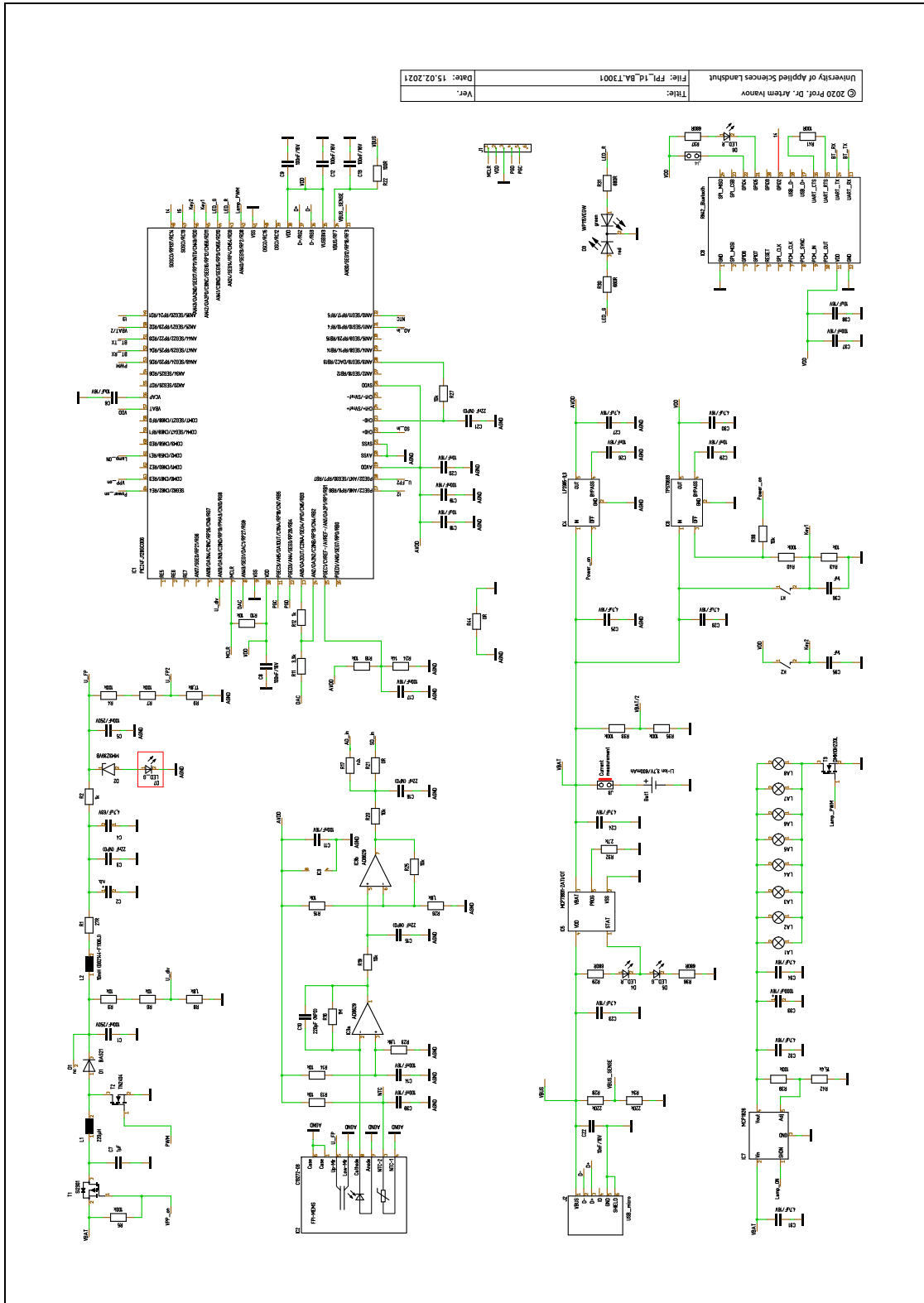
# Anhang



# A Bilder des mobilen Nah-Infrarot-Spektrometers



# B Schaltplan Leiterplatte



# C Sensor Final Inspection Sheet

FINAL INSPECTION SHEET 検査成績書		HAMAMATSU PHOTONICS K.K. SOLID STATE DIVISION 浜松ホトニクス株式会社 固体事業部	
Product name 製品名	MEMS-FPI Spectrum Sensor	Approved by 責任者	<i>Takeshi Kishikawa</i>
Type 型名	C13272-03	Inspected by 検査員	<i>Yumi Kurosawa</i>
Management code 管理コード	888549	Date inspected 検査日	20/08/05
Quantity 個数	1	Ambient temperature 測定環境温度	25℃

Inspection item 検査項目		Peak voltage at λ=1850nm <sup>※1</sup> ピーク電圧(λ=1850nm 時)	Peak voltage at λ=1550nm <sup>※1</sup> ピーク電圧(λ=1550nm 時)	Dark current at $V_R=0.5V$ ダーク電流 [nA]	Thermistor resistance サーミスタ抵抗値 [Ω]
Criteria 判定基準		≤ 32	≤ 42	≤ 100	9600 ≤ R ≤ 10400
Serial No.	DataMatrix input informaiton				
0H00016	C13272-03,0H00016	26.441	38.079	10.2	9949

※1 Test conditions : Incidence angle 0°, Photodetector NA 0.09, The incident spectral resolution (FWHM) Max.3nm  
検査条件 : 入射角度 0°, 受光素子 NA 0.09, 入射輝線分解能(FWHM) Max.3nm



# D Sensor Reference Datasheet

REFERENCE DATA SHEET リファレンスデータシート		HAMAMATSU PHOTONICS K.K. SOLID STATE DIVISION 浜松光子エレクトロニクス株式会社 固体事業部					
Product name 製品名	MEMS-FPI Spectrum Sensor	Type 型名	C13272-03				
Management code 管理コード	888549	Quantity 個数	1				
Serial No.	Sensor calibration coefficient data ※1,2,3 龍正:波長換算係数データ						
0H00016	C13272-03,0H00016	4.287583940E+32	-1.998767178E+26	1.077939304E+21	2.082576415E-06		
※1 Test conditions: Incidence angle 0°, Photodetector NA 0.09, The incident spectral resolution (FWHM) Max.3nm, T <sub>a</sub> =25°C 検査条件: 入射角0°, 受光素子NA 0.09, 入射波長分解能(FWHM) Max.3nm, T <sub>a</sub> =25°C							
※2 A transmission wavelength of a MEMS-FPI filter at 25 deg C can be estimated by substituting the a, b, c and g parameters for with the formula (4) in instruction manual, and then doing the (5) in instruction manual calculation. パラメータa,b,c,gを前述説明書の式(4)に代入し、式(5)を計算することにより、25°C時のMEMS-FPIフィルタの透過波長に対する印加電圧を推定することが出来ます。							
※3 This data is reference data. The value calculated from the above approximation formula doesn't guarantee the absolute wavelength. 本データは参考データです。上記近似式より算出される値は、絶対波長を保証するものではありません。							



REFERENCE DATA SHEET リアクティブセンサーシート		HAMAMATSU PHOTONICS K.K. SOLID STATE DIVISION 浜松光子学株式会社 固体事業部						
Product name 製品名	MEMS-FPI Spectrum Sensor	Type 型名	CI3272-03					
Management code 管理コード	888549	Quantity 個数	1					
Serial No.	DataMatrix input information	Sensor calibration coefficient data 電圧-波長換算係数データ						
	p	q	r	s	t	u	v	w
	7.954559615E+28	1.425654048E+23	3.751786275E+17	7.516623670E+10	-6.368411338E+27	8.040292393E+21	-2.816071871E+15	2.376181773E+07
<p>※1 Test conditions : Incidence angle 0°, Photodiode NA 0.09. The incident spectral resolution (FWHM) Max. 5nm 検査条件 : 入射角度 0°, 受光素子NA 0.09 , 入射線幅分解能(FWHM) Max. 5nm</p> <p>※2 The temperature compensation parameters (p, q, r, s, t, u, v and w) substituted for the formulas (18), (9~14) in instruction manual is to make the temperature compensation possible. 温度補正パラメータ(p, q, r, s, t, u, v, w)を取扱説明書の式(18), 式(9~14)に代入し, 式(15)にすることで, 温度補正も可能となります。</p> <p>※3 This data is reference data. The value calculated from the above approximation formula doesn't guarantee the absolute wavelength. 本データは参考データです。上記近似式より算出される値は, 絶対波長を保証するものではありません。</p>								



# E Sensor Calibration Coefficients

```
C13272-03 0H00016  
a0 -22196.108369884  
a1 0  
a2 528.467224641397  
a3 -56.6520803396309  
a4 2.72620604859094  
a5 -6.98338436448412E-02  
a6 9.3009066871528E-04  
a7 -5.09022224482345E-06
```

# F Erzeugung Steuerspannung

Arne Kulinna

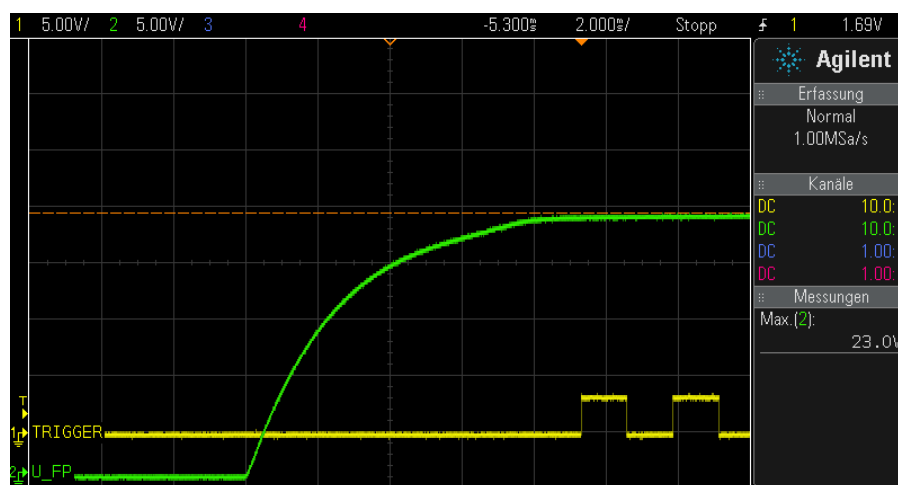
30.10.2020

## Erzeugung Steuerspannung

$U_{FP}$  ist die am Ausgang des DC/DC Step-Up Wandler erzeugte Spannung.  
Das Signal TRIGGER dient lediglich der zeitlichen Koordination und hat keinen analytischen Wert.

$t_0$  entspricht vertikaler Mittellinie.

Ist-Spannung: 0V; Soll-Spannung: 23,1V; Startzeit:  $t_0 - 4ms$



Ist-Spannung: 0V; Soll-Spannung: 37,7V; Startzeit:  $t_0 - 10ms$ .

Signalverläufe für  $t < -10ms$  sind testbedingt zu vernachlässigen.



# G Halogen Lamp Selection

## Halogen Lamp Selection

```

% The following voltages are measured at amplifier output
% U_{SDoff} is measured with U_{FP} = 0V
% U_{SDon} is measured with U_{FP} = 30.5V

% Distance between filament and sensor equals 40mm

% Lamp matrices [Type, U_Lamp, I_Lamp, U_SDoff, U_SDon]

Lamp = [
7362  5  112 580 680
7362  4  99  576 650
7362  3  86  576 625
7868  2.5 350 580 735
7868  2  308 580 687
7328  3  141 577 644
7328  4  165 580 690
7328  5  185 581 734
7361  3  43  575 607
7361  4  50  575 625
];

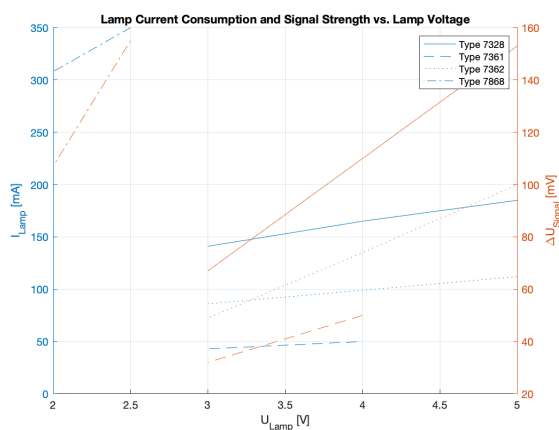
% Group lamps by type
GroupI = findgroups(Lamp(:,1));
Groups = splitapply(@(x){x}, Lamp, GroupI);

hold on
LegendText = strings(size(Groups));

for I = 1:size(Groups)
    Group = Groups{I};
    yyaxis left
    plot(Group(:, 2), Group(:, 3))
    yyaxis right
    plot(Group(:, 2), (Group(:, 5) - Group(:, 4)))
    LegendText(I) = sprintf('Type %d', Group(1, 1));
end

title('Lamp Current Consumption and Signal Strength vs. Lamp Voltage')
xlabel('U_{Lamp} [V]')
yyaxis left
ylabel('I_{Lamp} [mA]')
yyaxis right
ylabel('\Delta U_{Signal} [mV]')
legend(LegendText);
grid on
hold off

```





# H Temperature Calculations

## Temperature Calculations

### Thermistor Curve

```
% Define constants [Rmax, Rmin, A, B, C]
Constants = [
    300.3  96.1  0.003533480000000000 -7.87463000000000e-05  8.74259000000000e-07;
    96.1   32.9  0.001465160000000000  0.000185960000000000  2.32844000000000e-07;
    32.9   5.3   0.001093160000000000  0.000240113000000000  6.28813000000000e-08;
    5.3    1.1   0.001059550000000000  0.000244870000000000  5.14556000000000e-08
]
```

```
Constants = 4x5
    300.3000  96.1000  0.0035  -0.0001  0.0000
    96.1000  32.9000  0.0015  0.0002  0.0000
    32.9000  5.3000  0.0011  0.0002  0.0000
    5.3000   1.1000  0.0011  0.0002  0.0000
```

```
% Reference NTC resistance value for 25°C in kOhm
ReferenceExampleR = 10
```

```
ReferenceExampleR = 10
```

```
% True sensor NTC resistance value for 25°C in kOhm
SensorR = 9.949
```

```
SensorR = 9.9490
```

```
% Full range resistance values
R = 1.1:1:300.1;
```

```
% Calculate difference between reference and sensor at 25°C
ReferenceExampleT = SteinHartHart(Constants, ReferenceExampleR)
```

```
ReferenceExampleT = 25.0181
```

```
SensorT = SteinHartHart(Constants, SensorR)
```

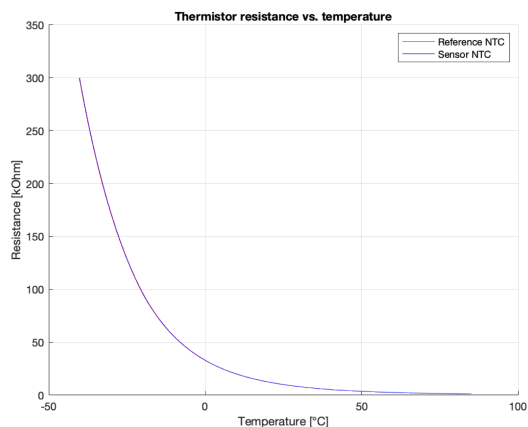
```
SensorT = 25.1346
```

```
DiffT = ReferenceExampleT - SensorT
```

```
DiffT = -0.1165
```

```
% Calculate full range temperature curve of reference NTC
FullCurveT = SteinHartHart(Constants, R);
% Calculate full range temperature curve of sensor NTC
SensorCurveT = FullCurveT - DiffT;
```

```
% Plot result
hold on
plot(FullCurveT, R, "r")
plot(SensorCurveT, R, "b")
title('Thermistor resistance vs. temperature')
xlabel('Temperature [°C]')
ylabel('Resistance [kOhm]')
legend('Reference NTC', 'Sensor NTC')
grid on
hold off
```



## Further calculations for microcontroller

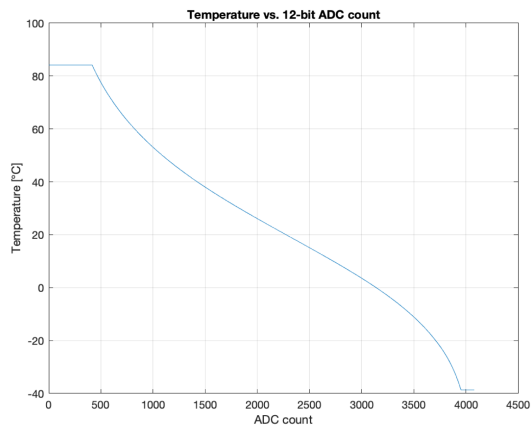
```

ADC = 0:16:4095;
% Calculate resistance from ADC value
ADCR = (-10 * ADC) ./ (ADC - 4096);
% Calculate temperatures from resistance
ADCT = SteinHartHart(Constants, ADCR);

% Saturate min and max values
ADCTmax = max(ADCT);
ADCTmaxI = find(ADCT == ADCTmax);
ADCTmin = min(ADCT);
ADCTminI = find(ADCT == ADCTmin);
ADCT(1:ADCTmaxI) = ADCTmax;
ADCT(ADCTminI:end) = ADCTmin;

% Plot result
plot(ADC, ADCT)
title('Temperature vs. 12-bit ADC count')
xlabel('ADC count')
ylabel('Temperature [°C]')
grid on

```



```

% Print Look-up-Table
LUKT = round(ADCT, 0)

```

```

LUKT = 1×256
    84    84    84    84    84    84    84    84    84    84    84    84    84    84    84    84    84    84    84    84

```

```

function TempDegC = SteinHartHart(Constants, R)
% Allocate memory
TempDegC = zeros(size(R));

for I = 1:size(R, 2)
% Current resistance value
Rn = R(I);
% Extract constants for current resistance value
Constant = Constants(Rn < Constants(:,1) & Rn >= Constants(:,2),:);
% Prevent error if value is outside of range
if isempty(Constant) == 0
% Extract constants for readability
A = Constant(:,3);
B = Constant(:,4);
C = Constant(:,5);
% Convert kOhm to Ohm
Rn = Rn * 1000;
% Calculate Steinhart-Hart equation
TempDegC(I) = (1./(A + B * log(Rn) + C * power(log(Rn), 3))) - 273.15;
end
end
end

```

# I Message Payload Content

```
/*
 * Message Payload Content
 * Author: Arne Kulinna
 *
 * Created on November 11, 2020, 01:24 PM
 */

Type 0x63
{
    {
        {
            u16 IEEE754DoubleLSB
            u16 IEEE754DoubleMSB

        } [40] SensorCoefficients
        // Respective sensor coefficients a - w and a0 - a7

        u16 MeasurementType
        // 0 -> Standard measurement
        // 1 -> Reference measurement

        u16 LampState
        // 0 -> Lamps off
        // 1 -> Lamps on

    }
    {
        u16 SamplePoints
        // Divider of sample points based on 1024
        // 1 -> 1024
        // 2 -> 512
        // 4 -> 256
        // 8 -> 128
        // 16 -> 64

        u8 AverageNumber
        // Number of measurements averages for final result
        // [1..32]

        u8 Delay
        // Delay counts between sample points
        // t = Delay * 1.28ms

    }
    {
        u32:20 Signal
        // Light intensity in fixed point ADC counts with factor 100

        u32:12 FPIVoltage
        // Fabri-Pérot Interferometer control voltage in ADC counts

        s8 Temperature
        // Temperature in degree celsius

        u8 Reserved

    } [64 - 1024]
    // Size depends on number of sample points
}
}
```

**Type 0x27**

```
{
  u16 BatteryValue
  // U_Bat = BatteryValue / 10
}
```

**Type 0x72 and 0x78**

```
{
  {
    u16 SamplePoints
    // Divider of sample points based on 1024
    // 1 -> 1024
    // 2 -> 512
    // 4 -> 256
    // 8 -> 128
    // 16 -> 64

    u8 AverageNumber
    // Number of measurements averaged for final result
    // Mind range [1..32]

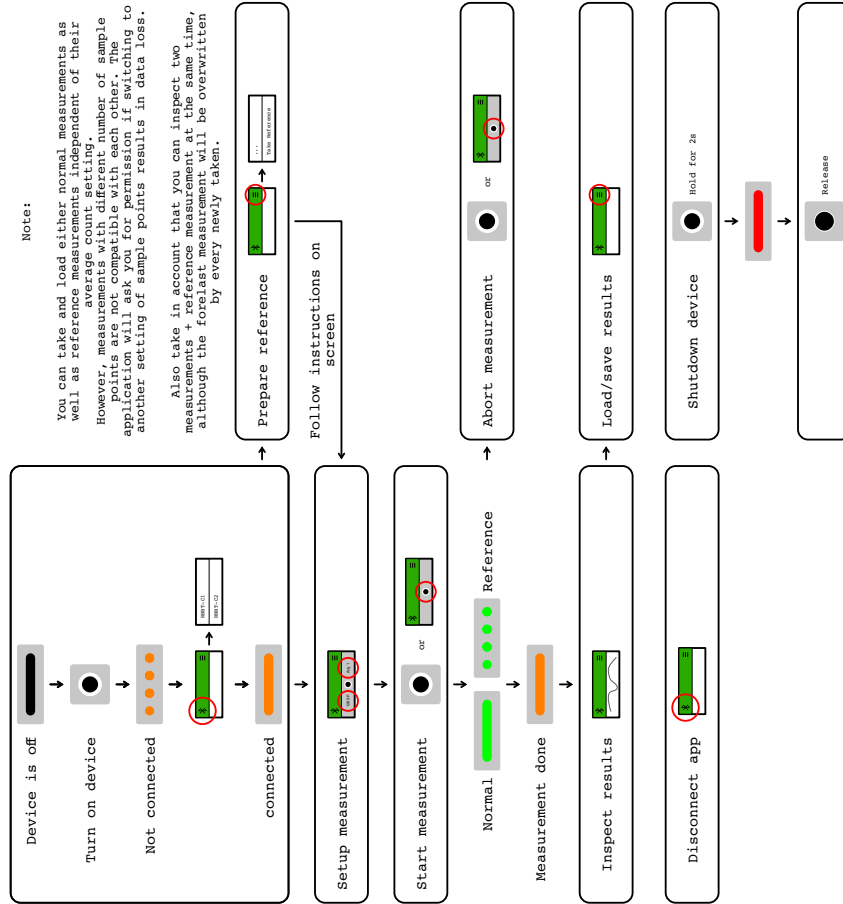
    u8 Delay
    // Delay counts between sample points
    // t = Delay * 1.28ms
  }
}
```

**Type 0x77**

```
{
  u16 SystemState
  // 0 -> SYS_OFF_RDY
  // 1 -> SYS_ERROR
  // 2 -> SYS_ON_DISC
  // 3 -> SYS_ON_CONN
  // 4 -> SYS_MEAS
}
```

# J Quick-Start User Guide

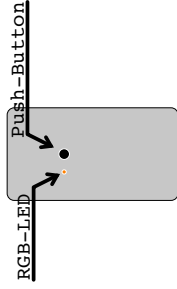
## Quick-Start User Guide



Note:

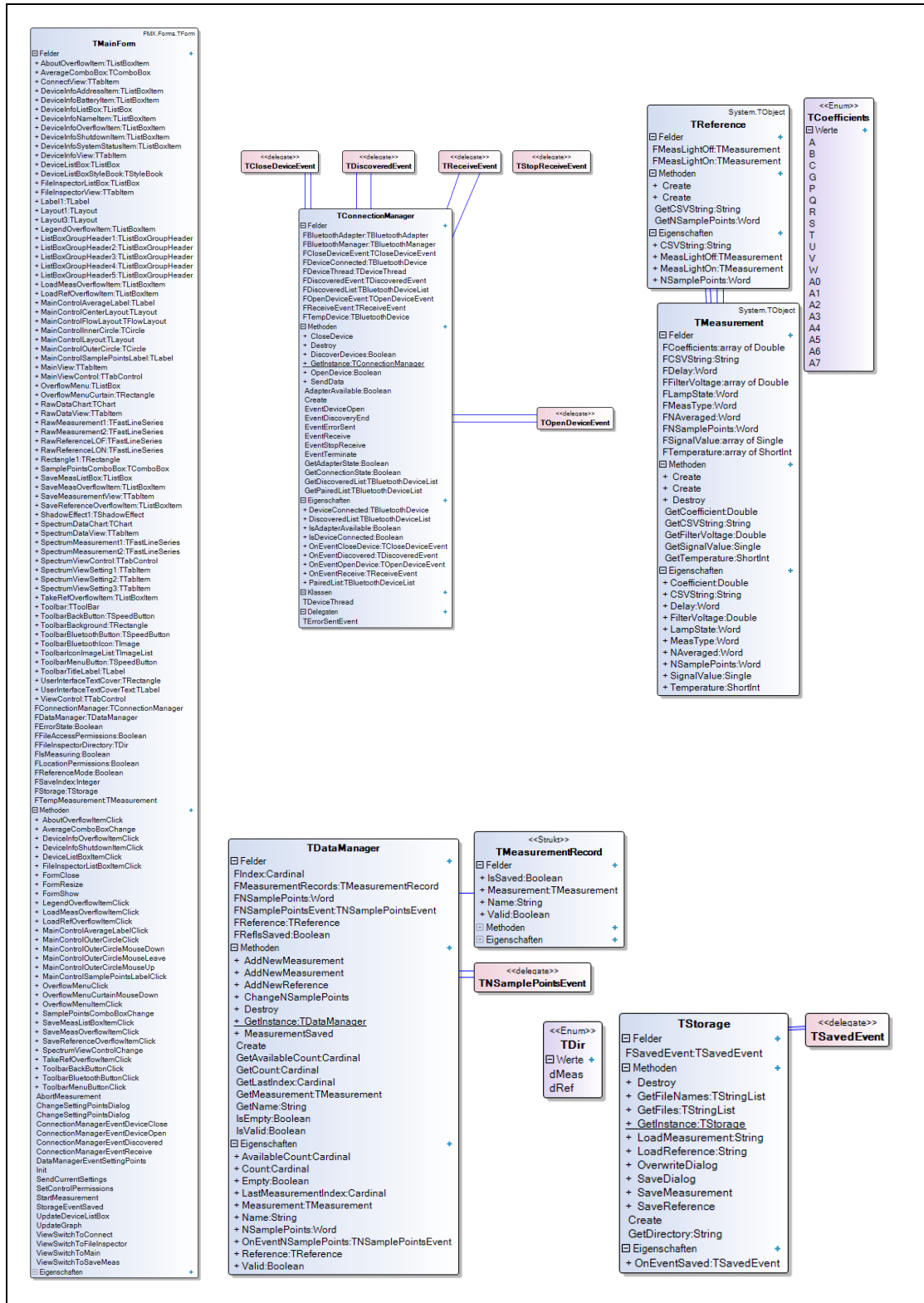
You can take and load either normal measurements, as well as reference measurements independent of their average count setting. However, measurements with different number of sample points are not compatible with each other. The application will ask you for permission if switching to another setting of sample points results in data loss.

Also take in account that you can inspect two measurements of reference measurement at the same time, although the former measurement will be overwritten by every newly taken.



RGB-LED Status	Description
● ● ● ●	Not connected
● ● ● ● ●	Connected, ready to measure
● ● ● ● ● ●	Active reference measurement
● ● ● ● ● ● ●	Active measurement
● ● ● ● ● ● ● ●	Error, low battery
● ● ● ● ● ● ● ● ●	Ready to shutdown

# K Klassen der Visualisierungssoftware



# L File Content Structure

Content structure of a „meas“ file. Variable data is red. Corresponding datatype is orange.

Row				
0	_BTAK	MEAS	20	
1	Info			
2	Type		0	
3	Lamp		0	
4				
5	Settings			
6	NSamplePoints		1024	[ 64   128   256   512   1024 ]
7	NAverage		1	[ 1 - 32 ]
8	Delay		0	[ 0 - 255 ]
9				
10	Coefficients			
11	a		-0.0698338436448412	{ Double Precision Floating Point Value }
12	b		-1.998767178E28	{ Double Precision Floating Point Value }
13	c		1.077939304E21	{ Double Precision Floating Point Value }
14	g		1	{ Double Precision Floating Point Value }
15	p		7.954559615E28	{ Double Precision Floating Point Value }
16	q		1.425654048E23	{ Double Precision Floating Point Value }
17	r		3.751786275E17	{ Double Precision Floating Point Value }
18	s		75166236700	{ Double Precision Floating Point Value }
19	t		-6.368411338E27	{ Double Precision Floating Point Value }
20	u		2.082576415E-6	{ Double Precision Floating Point Value }
21	v		-2.816071871E15	{ Double Precision Floating Point Value }
22	w		23761817.73	{ Double Precision Floating Point Value }
23	a0		-22196.108369884	{ Double Precision Floating Point Value }
24	a1		0	{ Double Precision Floating Point Value }
25	a2		528.467224641397	{ Double Precision Floating Point Value }
26	a3		-56.6520803396309	{ Double Precision Floating Point Value }
27	a4		2.72620604859094	{ Double Precision Floating Point Value }
28	a5		8.040292396E21	{ Double Precision Floating Point Value }
29	a6		0.00093009066871528	{ Double Precision Floating Point Value }
30	a7		-5.09022224482345E-6	{ Double Precision Floating Point Value }
31				
32	Data			
33	Signal	U_FP	Temp	
34		4278	23.0581086914062	24
...				{ Single Precision Floating Point Value } { Double Precision Floating Point Value } [ -128 - 127 ]
Max. 1058		4317	37.855125	24

Content structure of a „ref“ file consists of two „meas“ files append to the reference header.

Row				
0	_BTAK	REF	20	
1	_BTAK	MEAS	20	
	Info			
2	Type		1	
3	Lamp		1	
...	...	...	...	...
	_BTAK	MEAS	20	
	Info			
	Type		1	
	Lamp		0	
...	...	...	...	...
Max. 2117				

# Literaturverzeichnis

- [1] Crc polynomial zoo. <https://users.ece.cmu.edu/~koopman/crc/crc16.html>. Zuletzt zugegriffen: 26.02.2021.
- [2] Mems-fpi questions & answers — hamamatsu photonics. <https://hub.hamamatsu.com/us/en/ask-engineer/MEMS-FPI-questions-&-answers/index.html>. Zuletzt zugegriffen: 02.02.2021.
- [3] Mems-fpi spectroscopic module — hamamatsu photonics. <https://www.hamamatsu.com/eu/en/product/photometry-systems/spectroscopic-modules/index.html>. Zuletzt zugegriffen: 02.04.2021.
- [4] Nir spectroscopy / trinamix. <https://trinamixsensing.com/nir-spectroscopy/>. Zuletzt zugegriffen: 02.04.2021.
- [5] Qt success stories - automotive. <https://resources.qt.io/customer-stories-automotive>. Zuletzt zugegriffen: 02.02.2021.
- [6] Rad studio - Überblick - embarcadero website. <https://www.embarcadero.com/de/products/rad-studio>. Zuletzt zugegriffen: 02.02.2021.
- [7] Steema — teechart chart components for vcl/fmx - delphi, c++builder, rad studio, firemonkey. <https://www.steema.com/product/vcl>. Zuletzt zugegriffen: 03.02.2021.
- [8] 16-Bit Sigma-Delta A/D Converter, November 2012. Zuletzt zugegriffen: 18.08.2020.
- [9] SERIAL PORT PROFILE, Juli 2012. Revision V12, Zuletzt zugegriffen: 22.02.2021.
- [10] Bluetooth Data Module Command Reference & Advanced Information User's Guide, März 2013. Version 1.0r, Zuletzt zugegriffen: 18.08.2020.
- [11] Universal Asynchronous Receiver Transmitter (UART), August 2013. Zuletzt zugegriffen: 18.08.2020.



- 
- [12] PIC24FJ128GC010 FAMILY Datasheet, November 2016. Zuletzt zugegriffen: 18.08.2020.
- [13] MEMS-FPI Spectrum Sensor C12272, C13272-02/03, C14273 Instruction Manual, Dezember 2019. Revision B.
- [14] Paul E. Black. Dictionary of algorithms and data structures. <https://www.nist.gov/dads/HTML/HammingDistance.html>, 2006. Zuletzt zugegriffen: 06.04.2021.
- [15] Elena Both. Scio: The unsteady journey of a kickstarter darling that promised big discoveries. <https://nocamels.com/2019/03/scio-kickstarter-darling-promises-molecular-sensor/>. Zuletzt zugegriffen: 02.04.2021.
- [16] Steven D. Brown, Romà Tauler, and Beata Walczak. Comprehensive Chemometrics. Elsevier, 2009.
- [17] Liu Chang. Foundations of MEMS. Pearson Education, 2012.
- [18] Martin Ebermann, Norbert Neumann, Karla Hiller, Mario Seifert, Marco Meinig, and Steffen Kurth. Tunable mems fabry-pérot filters for infrared microspectrometers: A review. [https://www.infratec.de/downloads/en/company/publications/2017\\_spie-pw-ebermann-tunable-mems.pdf](https://www.infratec.de/downloads/en/company/publications/2017_spie-pw-ebermann-tunable-mems.pdf), 2017. Zuletzt zugegriffen: 06.07.2020.
- [19] Hamamatsu Photonics K.K. MEMS-FPI spectrum sensor - Technical note, September 2020. Zuletzt zugegriffen: 30.03.2021.
- [20] Hamamatsu Photonics K.K. MEMS-FPI spectrum sensor C13272-03, Februar 2020. Zuletzt zugegriffen: 06.07.2020.
- [21] Digital Guide IONOS. Qt: Das c++-framework zur entwicklung plattformübergreifender software. <https://www.ionos.de/digitalguide/server/knowhow/qt/>. Zuletzt zugegriffen: 02.02.2021.
- [22] Philip Koopman. 32-bit cyclic redundancy codes for internet applications. <https://ieeexplore.ieee.org/document/1028931>, 2002. Zuletzt zugegriffen: 26.02.2021.

- 
- [23] Hans Kuzmany. Solid-State Spectroscopy. Springer, 1998.
- [24] Cornelius T. Leondes. MEMS/NEMS Handbook. Springer, 2006.
- [25] Werner Schmidt. Optische Spektroskopie. WILEY-VCH, 2000.
- [26] IEEE Computer Society. Ieee standard for ethernet. [http://www.onetel.de/wp-content/uploads/2016/11/802.3-2015\\_SECTION1.pdf](http://www.onetel.de/wp-content/uploads/2016/11/802.3-2015_SECTION1.pdf), 2015. Zuletzt zugegriffen: 25.03.2021.
- [27] IEEE Computer Society. Ieee standard for floating-point arithmetic. <https://ieeexplore.ieee.org/servlet/opac?punumber=8766227>, 2019. Zuletzt zugegriffen: 31.03.2021.
- [28] James M. Thompson. Infrared Spectroscopy. Pan Stanford Publishing, 2018.
- [29] Gerhard Wiegand. Gasmesstechnik in Theorie und Praxis. Springer Vieweg, 2016.