



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

# Bachelorthesis

André Marianovich

Integration eines digitalen MEMS-Mikrofons in  
ein photoakustisches Spektrometer

*Fakultät Technik und Informatik  
Department Informations- und  
Elektrotechnik*

*Faculty of Engineering and Computer Science  
Department of Information and  
Electrical Engineering*

André Marianovich  
Integration eines digitalen MEMS-Mikrofons in  
ein photoakustisches Spektrometer

Bachelorthesis eingereicht im Rahmen der Bachelorprüfung  
im Studiengang Informations- und Elektrotechnik  
am Department Informations- und Elektrotechnik  
der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr.Ing. Karl-Ragnar Riemschneider  
Zweitgutachter : Prof. Dr.Ing. Marcus Wolff

Abgegeben am 25. April 2013

**André Marianovich**

**Thema der Bachelorthesis**

Integration eines digitalen MEMS-Mikrofons in ein photoakustisches Spektrometer

**Stichworte**

Photoakustische Spektroskopie, Goertzel-Algorithmus, ARM Cortex M3, phasenempfindliche Signalverarbeitung,

**Kurzzusammenfassung**

Das Verfahren der photoakustischen Spektroskopie bietet durch immer kleinere und leistungsfähigere Mikroelektronik neue Anreize zur Verwendung in vielfältigen Bereichen der Gasanalytik. Ein spezieller Bereich stellt dabei der Einsatz in der Medizintechnik und hier in der Früherkennung beispielsweise von Lungenkrebs dar. Um dieses Messverfahren in medizintechnischen Apparaturen günstiger und massentauglich produzieren zu können, bedarf es der Optimierung hin zu kleineren Analysatoren. Diese Arbeit befasst sich mit der Entwicklung eines photoakustischen Mikrocontroller gesteuerten Messsystems auf der Basis eines digitalen MEMS-Mikrofons und der Erprobung neuer Signalverarbeitungs Alternativen

**André Marianovich**

**Title of the paper**

Integration of a digital MEMS based microphone in a photoacoustic spectrometer

**Keywords**

photoacoustic spectroscopy, goertzel algorithm, ARM Cortex M3, phase-sensitive signal processing

**Abstract**

The method of photoacoustic spectroscopy offers by ever smaller and more powerful microelectronics new incentives for use in various areas of gas analysis. A special area is represented by the use in medical and here especially in the early detection of lung cancer. To produce this measurement method suitable for mass production and cheaper in medical equipment, optimization is required to smaller analyzers. This work deals with the development of a microcontroller-controlled photoacoustic measurement system based on a digital MEMS microphone and the testing of new signal processing alternatives

# Inhaltsverzeichnis

<b>Tabellenverzeichnis</b>	<b>6</b>
<b>Abbildungsverzeichnis</b>	<b>7</b>
<b>1 Einführung</b>	<b>9</b>
<b>2 Photoakustische Spektroskopie</b>	<b>10</b>
2.1 Photoakustischer Effekt . . . . .	10
2.2 Messaufbau . . . . .	12
2.3 Kondensator- und Elektret-Mikrofone . . . . .	14
<b>3 Hardware</b>	<b>17</b>
3.1 MEMS Mikrofon ADMP441 . . . . .	17
3.1.1 Eigenschaften MEMS-basierter digitaler Mikrofone . . . . .	17
3.1.2 Vergleich der Spezifikationen von analogen und digitalen Mikrofonen	18
3.1.3 Betriebsparameter des ADMP441 . . . . .	19
3.2 Stellaris EKI-LM3S9B92 . . . . .	21
<b>4 Signalverarbeitung</b>	<b>24</b>
4.1 Phasenempfindlicher Verstärker (Lock-In) . . . . .	24
4.2 Fourier-basierte Spektralanalyse . . . . .	26
4.3 Goertzel-Algorithmus . . . . .	29
4.3.1 Herleitung . . . . .	29
4.3.2 Eigenschaften . . . . .	33
4.3.3 Stabilität . . . . .	35
<b>5 Implementierung der Firmware</b>	<b>36</b>
5.1 Konfiguration der Hardware . . . . .	36
5.1.1 Systemtakt . . . . .	36
5.1.2 UART . . . . .	37
5.1.3 I <sup>2</sup> S . . . . .	37
5.1.4 uDMA . . . . .	38
5.1.5 Timer . . . . .	39
5.2 Firmware . . . . .	40

---

5.2.1	USB Mikrofon Firmware . . . . .	40
5.2.2	Goertzel-Algorithmus Firmware . . . . .	41
<b>6</b>	<b>MEMS-basierte Photoakustik</b>	<b>46</b>
6.1	Experimenteller Aufbau . . . . .	46
6.2	Versuchsdurchführung . . . . .	47
6.2.1	Allgemeines und zur Lichtquelle . . . . .	47
6.2.2	Anmerkungen zur Messung mit der USB Mikrofon Firmware . . . . .	48
6.2.3	Anmerkungen zur Messung mit der Goertzel-Algorithmus Firmware . . . . .	50
6.3	Experimentelle Ergebnisse . . . . .	52
6.3.1	Messungen mit der USB Mikrofon Firmware . . . . .	52
6.3.2	Messungen mit der Goertzel-Algorithmus Firmware . . . . .	55
6.4	Messtechnische Probleme . . . . .	58
6.5	Diskussion der Implementierungsarten des Goertzel-Algorithmus . . . . .	60
<b>7</b>	<b>Zusammenfassung und Ausblick</b>	<b>62</b>
7.1	Nachweisempfindlichkeit . . . . .	63
7.2	Weiterführende Untersuchungen . . . . .	63
	<b>Literaturverzeichnis</b>	<b>64</b>
	<b>Anhang</b>	<b>67</b>

# Tabellenverzeichnis

6.1	Geräteliste . . . . .	46
6.2	Prüfgas . . . . .	47
6.3	Mittelwert und Standardabweichung der Goertzel-Algorithmus Firmware in Abhängigkeit der Sequenzlänge $N$ . . . . .	50
6.4	Mittelwerte der Goertzel-Algorithmus Firmware und des Goertzel- Algorithmus in Matlab für $N = 2000$ und des Lock-In Verstärkers sowie die gemessene Wellenlänge, optische Leistung und Chopper-Frequenz. Zelldruck: 1015 mbar . . . . .	56

# Abbildungsverzeichnis

2.1	Anregung des energetischen Zustandes durch ein Photon. . . . .	11
2.2	Beispielhafter Verlauf eines photoakustischen Signals, gemessen mit einem analogen Elektret-Mikrofon und Mikrofonvorverstärker. . . . .	12
2.3	Typischer Messaufbau der photoakustischen Spektroskopie. . . . .	13
2.4	Modifizierte Messaufbau zur photoakustischen Spektroskopie. . . . .	14
2.5	Prinzipschaltung eines Kondensator-Mikrofons mit Impedanzwandler, frei nach [1] . . . . .	15
2.6	Frequenzgang des Primo EM158 [2] . . . . .	16
3.1	Blockschaltbild des ADMP441. Quelle: Analog Devices . . . . .	19
3.2	Frequenzgang des digitalen Tiefpassfilters des ADMP441. Quelle: [14] . . . . .	20
3.3	Datentransfer der I <sup>2</sup> S Schnittstelle. Quelle: [3] . . . . .	21
3.4	Überblick über die Komponenten des LM3S9B92. Quelle: Texas Instruments . . . . .	22
4.1	Funktionsblöcke eines Lock-In Verstärkers. . . . .	25
4.2	DFT Spektrum als Filterbank nach [4]. . . . .	27
4.3	Signalflussplan des Goertzel-Algorithmus nach Gleichung 4.3.10. . . . .	31
4.4	Signalflussplan des modifizierten Goertzel-Algorithmus nach Gleichung 4.3.16 und 4.3.17. . . . .	32
4.5	Leistungsfrequenzgang von 4.3.26. Die Abtastfrequenz ist 100 Hz . . . . .	34
5.1	Programmflussplan der Goertzel-Algorithmus Firmware . . . . .	42
6.1	Im Heinrich-Blasius Institut entstandene Leiterplatine mit dem montierten MEMS-Mikrofon ADMP441 (rot umrandet) und dem Kontakt für das Chopper-Signal (blau umrandet) . . . . .	47
6.2	Ergebnisse des Goertzel-Algorithmus in Abhängigkeit der Sequenzlänge N . . . . .	51
6.3	Gegenüberstellung der Amplitudenspektren beider Signalpfade bei 134,90mW optischer Ausgangsleistung und einer Wellenlänge von 3315,837nm. Abtastfrequenz $F_s = 8kHz$ , Länge der FFT $N = 8000$ , nicht normiert. . . . .	52
6.4	Gegenüberstellung der Amplitudenspektren beider Signalpfade ohne Laseranregung. Abtastfrequenz $F_s = 8kHz$ , Länge der FFT $N = 8000$ , nicht normiert. . . . .	53

---

6.5	Gegenüberstellung der Amplitudenspektren beider Signalpfade bei $134,90mW$ optischer Ausgangsleistung und einer Wellenlänge von $3315,837nm$ . Abtastfrequenz $F_s = 8kHz$ , Länge der FFT $N = 8000$ , nicht normiert. . . . .	54
6.6	Frequenzgang des Goertzel-Filters (rot) mit realen Messdaten (blau), $N = 2000$ , normiert . . . . .	57
6.7	Frequenzgang des Goertzel-Filters (rot) mit realen Messdaten (blau), $N = 8000$ , normiert . . . . .	58
6.8	Schwankung der Chopper-Frequenz . . . . .	59
6.9	Unkontrollierte Schwankungen am Ausgang des Goertzel-Filters in einer Festkomma-Implementierung . . . . .	61



# 1 Einführung

Im Heinrich-Blasius-Institut für Physikalische Sensorik wird ein Forschungsprojekt durchgeführt, dessen Ziel die Entwicklung eines Analysators für flüchtige organische Komponenten mit extrem hoher Nachweisempfindlichkeit ist. Der Analysator nutzt das Messprinzip der photoakustischen Spektroskopie (PAS). Dieses Verfahren basiert auf der Absorption modulierter elektromagnetischer Strahlung und der Umwandlung der absorbierten Energie in eine akustische Welle. Dieses Schallsignal wurde bisher mit einem Elektret-Kondensatormikrofon gemessen und anschließend mit Hilfe der Lock-In-Technik phasensensitiv detektiert.

Ziel dieser Arbeit ist es, das Elektret-Mikrofon durch ein digitales Micro-Electro-Mechanical-Systems (MEMS) Mikrofon zu ersetzen. Zum Einsatz kommt das ADMP441 von Analog Devices. Durch das neuartige digitale Mikrofon entfällt der bisherige Mikrofonvorverstärker. Anstelle des Lock-In Verstärkers soll die Arbeit der Signalverarbeitung von einem Mikrocontroller übernommen werden.

Grundsätzlich ergeben sich dadurch zwei Teilaspekte innerhalb dieser Arbeit. Zum einen soll das ADMP441 mit dem bisher eingesetzten, analogen Elektret-Mikrofon verglichen und hinsichtlich seiner Eignung für die Verwendung innerhalb der photoakustischen Spektroskopie untersucht werden. Zum anderen soll nach einer möglichen Alternative für den Lock-In Verstärker gesucht werden.

Innerhalb der Einarbeitungs- und Konzeptentwicklungsphase hat sich herausgestellt, dass der Ansatz, mit einem MEMS-Mikrofon ein photoakustisches Messsystem auf der Basis eines Mikrocontrollers aufzubauen, bereits in ähnlicher Form realisiert wurde. Die Bachelor-Thesis zur *"Messung von Kohlenstoffpartikeln mittels Photoakustik"* der Schweizer Studenten Michael Glettig und Benjamin Wyrch sah ebenfalls die Verwendung des ADMP441 von Analog Devices und einen ARM Cortex M3 basierten Mikrocontroller vor.

Als Konsequenz wurde für die Signalverarbeitung ein alternatives Konzept zu der Lock-In-Technik gesucht und mit dem Goertzel-Algorithmus gefunden.

## 2 Photoakustische Spektroskopie

Die photoakustische Spektroskopie ist ein Laserspektroskopie-Verfahren, dem der photoakustische Effekt zugrunde liegt, welcher erstmals von Alexander Graham Bell im Jahr 1880 beschrieben wurde [5, S. 7]. Sie beruht auf der Absorption elektromagnetischer Strahlung durch die untersuchte Probe und besitzt eine äußerst hohe Nachweisempfindlichkeit von geringen Konzentrationen einer Molekülart innerhalb eines Gasgemisches. So ist der Nachweis einer Konzentration im Bereich von  $10^{-9}$  (parts per billion, ppb) möglich [6, S. 272].

Nachfolgend sollen einführend die Grundlagen des photoakustischen Effekts sowie der grundsätzliche Messaufbau in der photoakustischen Gasanalyse beschrieben werden. Anschließend wird auf die Eigenschaften des bisher zum Einsatz kommenden Elektret-Mikrofons eingegangen, da dieses einen Gegenstand dieser Arbeit darstellt.

### 2.1 Photoakustischer Effekt

Ein in einer photoakustischen Zelle (kurz Messzelle oder Zelle) befindliches Gas wird mit Laserlicht der Wellenlänge  $\nu$  bestrahlt. Entspricht die Wellenlänge des Lasers der Absorptionslinie des Gases in der Zelle, so wird durch die Absorption eines Photons mit der Energie

$$E_p = h\nu = \frac{h \cdot c_0}{\lambda} \quad (2.1.1)$$

das Gasmolekül von seinem Grundzustand  $E_0$  in einen energetisch höheren Zustand  $E_1$  angeregt [5, S. 15]:

$$\Delta E = E_1 - E_0 = h\nu \quad (2.1.2)$$

Dabei sind  $h = 6,626 \cdot 10^{-34} Js$  das Plancksche Wirkungsquantum,  $c_0 = 2,9979 \cdot 10^8 \frac{m}{s}$  die Lichtgeschwindigkeit und  $\lambda$  die Wellenlänge im Vakuum.

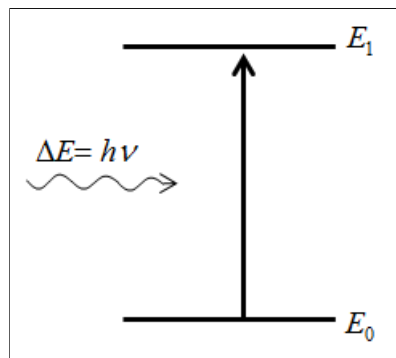


Abbildung 2.1: Anregung des energetischen Zustandes durch ein Photon.

Für die angeregten Moleküle gibt es vier Möglichkeiten in den Grundzustand zurückzukehren [5, S. 15]:

1. Bei dem Übergang vom angeregten in den Grundzustand kann das Molekül wieder ein Photon gleicher Frequenz emittieren
2. Durch das eingestrahlte Laserlicht wird eine chemische Reaktion ausgelöst (Photochemie)
3. Das angeregte Molekül kann mit einem Molekül derselben Art im Grundzustand kollidieren und so seine Energie über Stoßprozesse weitergeben
4. Im vierten Fall weiten sich die Stoßprozesse auf alle weiteren Moleküle aus, so dass die kinetische Gesamtenergie des Gases in der photoakustischen Zelle steigt. Durch den erhöhten Anstieg der kinetischen Energie des Gases, kommt es zu einer Erhöhung der Temperatur und damit zu einem Druckanstieg [6, S. 271]

Die letzten beiden Fälle basieren auf der Anregung der Vibrationszustände, was vorwiegend bei Lasern mit Wellenlängen im Infrarot-Bereich vorkommt [7, S. 80], wobei speziell der letzte Fall in der PAS genutzt wird.

Wird der Laser in seiner Intensität moduliert, so kann eine periodische Druckschwankung innerhalb der Zelle erzeugt werden, die wiederum mit einem Mikrofon gemessen werden kann [5, S. 16]. Diese Druckschwankung stellt das photoakustische Signal dar und besitzt in der Regel einen sinusförmigen Verlauf mit additivem Rauschen,

$$s(t) = s_0(t) \cdot \cos(\omega t + \varphi) + n(t). \quad (2.1.3)$$

Abbildung 2.2 zeigt die Aufnahme eines photoakustischen Signals mit einem Kondensator-Mikrofon nach dem Mikrofonvorverstärker bei einer Modulationsfrequenz von 2,586 kHz. Das Signal-Rausch-Verhältnis (engl. signal-to-noise ratio, SNR) lässt einen sinusförmigen Signalverlauf noch relativ klar erkennen.

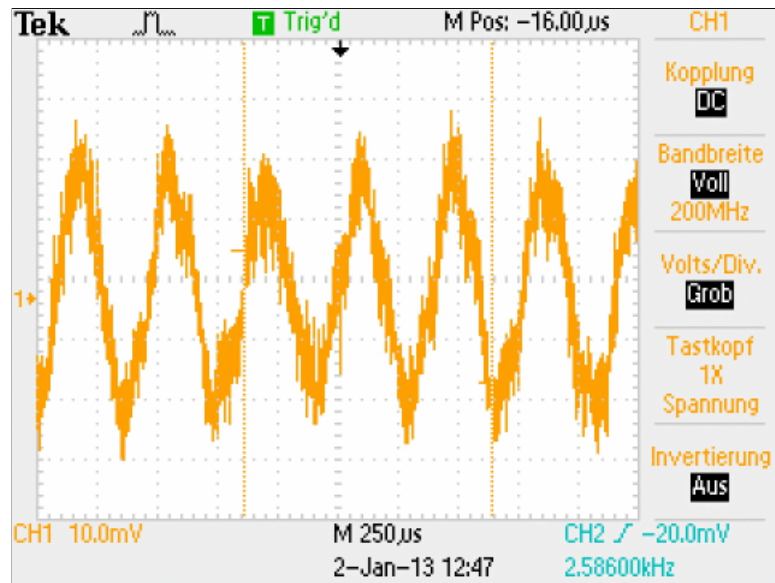


Abbildung 2.2: Beispielhafter Verlauf eines photoakustischen Signals, gemessen mit einem analogen Elektret-Mikrofon und Mikrofonvorverstärker.

Die Signalstärke ist dabei von mehreren Faktoren abhängig. Zu nennen sind die Teilchendichte  $N_i$  der absorbierenden Moleküle im Grundzustand, die mittlere Laserleistung  $P_L$  und die Empfindlichkeit  $R$  des Mikrofons [6, S. 272]. Des Weiteren besitzt die Geometrie der Messzelle einen Einfluss auf die Signalstärke. So kann eine Signalverstärkung vom Faktor zwei erreicht werden, wenn der Laser mit der Eigenresonanzfrequenz der Zelle moduliert wird [6, S. 272].

## 2.2 Messaufbau

In Abbildung 2.3 ist der gegenwärtige Messaufbau, wie er grundsätzlich in der photoakustischen Spektroskopie anzutreffen ist, schematisch dargestellt.

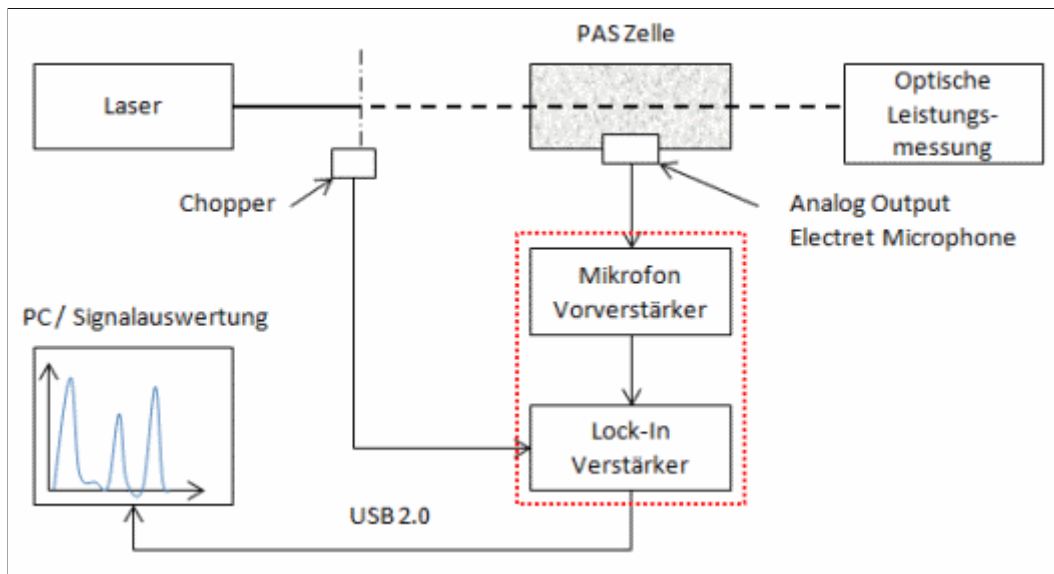


Abbildung 2.3: Typischer Messaufbau der photoakustischen Spektroskopie.

Das Licht des durch ein Chopper-Rad in der Intensität mit der Resonanzfrequenz der Messzelle modulierten Lasers tritt durch ein Fenster in die Zelle ein. Innerhalb der Zelle wird ein kleiner Teil der Laserphotonen von dem Probegas, auf dessen Spektrallinie die Wellenlänge des Lasers abgestimmt ist, absorbiert. Der Großteil des Laserstrahls tritt am Ende der Zelle durch ein zweites Fenster aus, wo ein optisches Leistungsmessgerät die Ausgangsleistung des Lasers misst. Bei der Untersuchungen von geringen Konzentrationen kann man annehmen, dass die hinter der Zelle gemessene Leistung in etwa der Laserleistung entspricht, die in die Zelle gestrahlt wird [5, S. 61].

Neben der Messzelle mit dem integrierten Mikrofon, samt Mikrofonvorverstärker, ist der abgebildete phasenempfindliche Verstärker, auch Lock-In Verstärker genannt, dessen Funktionsweise in Abschnitt 4.1 ausführlich beschrieben wird, der wichtigste Teil an diesem Messaufbau. Er dient dazu das photoakustische Signal zu detektieren, auch wenn dessen Amplitude im Rauschteppich verschwindet. Zur Detektion nutzt der Lock-In die Frequenz des Chopper-Rad als Referenzsignal.

Im Rahmen dieser Arbeit soll der in Abbildung 2.3 rot umrandete Teil des Messaufbaus durch einen Mikrocontroller ersetzt werden, so dass ein modifizierter Messaufbau wie in Abbildung 2.4 entsteht.

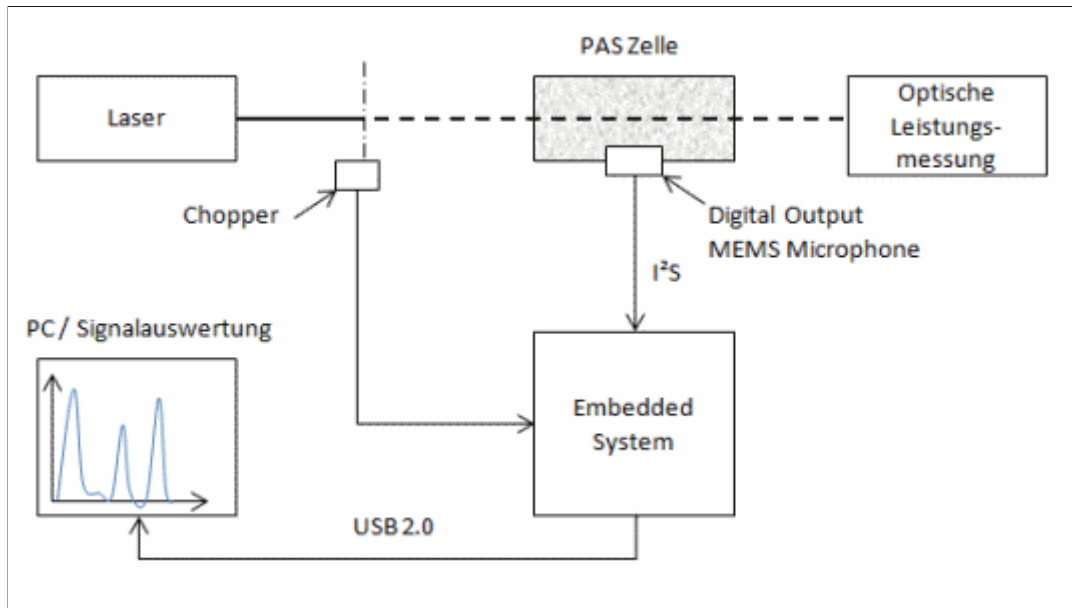


Abbildung 2.4: Modifizierter Messaufbau zur photoakustischen Spektroskopie.

### 2.3 Kondensator- und Elektret-Mikrofone

In der photoakustischen Spektroskopie werden als Sensoren zur Messung der Druckschwankung innerhalb der Messzelle vorwiegend analoge und niederfrequente Schallwandler eingesetzt, die auf dem elektrostatischen Prinzip beruhen [5, S. 61]. Dies sind Kondensatormikrofone, die einem Plattenkondensator nachempfunden sind, wobei die eine Elektrode als dünne elastische Membran ausgeführt wird [8, S. 259]. Der aufgenommene Schalldruck führt zu einer Auslenkung der Membran, was zu einer Änderung des Plattenabstands führt.

Das Verhalten der Kapazität des Kondensator-Mikrofons lässt sich nach [1, S. 352f] herleiten. Die Kapazität eines Plattenkondensators ist umgekehrt proportional zum Abstand  $d$  der Platten.

$$C_0 \sim \frac{1}{d} \quad (2.3.1)$$

Durch die Auslenkung der Membran verringert sich der Plattenabstand um  $\Delta d$  und es ändert sich die Kapazität zu

$$C \sim \frac{1}{d - \Delta d} \quad (2.3.2)$$

Setzt man Gleichung 2.3.2 und 2.3.1 zueinander ins Verhältnis, dann ergibt sich der folgende Ausdruck:

$$C = C_0 \frac{d}{d - \Delta d} = C_0 \frac{1}{1 - \frac{\Delta d}{d}} \quad (2.3.3)$$

Die kapazitive Änderung alleine führt aber noch nicht zu einem messbaren Ausgangssignal. Dazu muss der Kondensator über eine Versorgungsspannung vorgespannt werden. In der Tontechnik bezeichnet man diese als Phantomspeisung [8, S. 261].

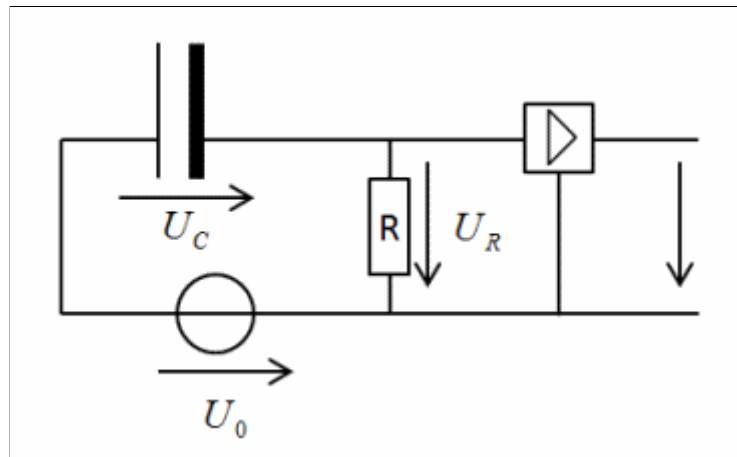


Abbildung 2.5: Prinzipschaltung eines Kondensator-Mikrofons mit Impedanzwandler, frei nach [1]

Damit es bei einer Kapazitätsänderung auch zu einer messbaren Spannungsänderung kommt und kein Ausgleichsstrom fließt, muss  $R \gg 1/j\omega C_0$  gelten. Bei dieser Dimensionierung verhält sich der Kondensator etwa so, wie im Leerlauf. Zusammen mit diesem Lastwiderstand bildet das Mikrofon einen CR-Hochpass, dessen Zeitkonstante gerade so ausgelegt wird, dass der tiefste aufzunehmende Ton und damit die langsamste Spannungsänderung über den Kondensator gerade nicht zu einem Ausgleichsstrom führt [1, S. 353f]

$$\omega \gg \frac{1}{RC_0} \quad (2.3.4)$$

Diese Dimensionierung legt daher auch die untere Grenzfrequenz des Mikrofons fest. Um das Signal des Mikrofons sinnvoll und ohne Verzerrungen abgreifen zu können, wird ein Impedanzwandler benötigt, der die hochohmige Mikrofonschaltung an nachgeschaltete Kabel und Schaltungen anpasst. Der Impedanzwandler besitzt in der Regel die Verstärkung 1

[8, S.260]. Die Ausgangsspannung lässt sich mit der Kondensatorspannung

$$U_C = \frac{Q}{C} = \frac{Q}{C_0} \left(1 - \frac{\Delta d}{d}\right) \quad (2.3.5)$$

entsprechend zu

$$U = U_R = U_0 - U_C \quad (2.3.6)$$

bestimmen. Ganz ohne eine Versorgungsspannung kommen so genannte Elektret-Kondensatormikrofone aus, deren Membran aus einem elastischen Elektret besteht, das eine permanente Ladung zur Polarisation des Kondensators besitzt [8, S.261]. Im weiteren Text sollen Kondensator- und Elektret-Mikrofone synonym verwendet werden.

Ein solches analoges Elektret-Mikrofon kommt in dem bisherigen Messaufbau, wie er in Abbildung 2.2 skizziert wurde, zum Einsatz. Speziell handelt es sich dabei um das EM158 von Primo. Es besitzt ein SNR von  $74\text{dB}$  (bei  $1\text{kHz}$  und  $1\text{Pa}$  gemessen) und eine Sensitivität von  $-32\text{dB} \pm 3\text{dB}$  (siehe dazu Abschnitt 3.1.2).

Der Frequenzgang des Primo EM158 ist in Abbildung 2.6 zu sehen. Ihm ist zu entnehmen, dass das Übertragungsverhalten des Schalldrucks in dem Bereich von  $2\text{kHz}$  bis  $3\text{kHz}$  noch relativ flach verläuft. In diesem Bereich wird, wie sich später noch zeigen wird, vorwiegend das photoakustische Signal aufgenommen.

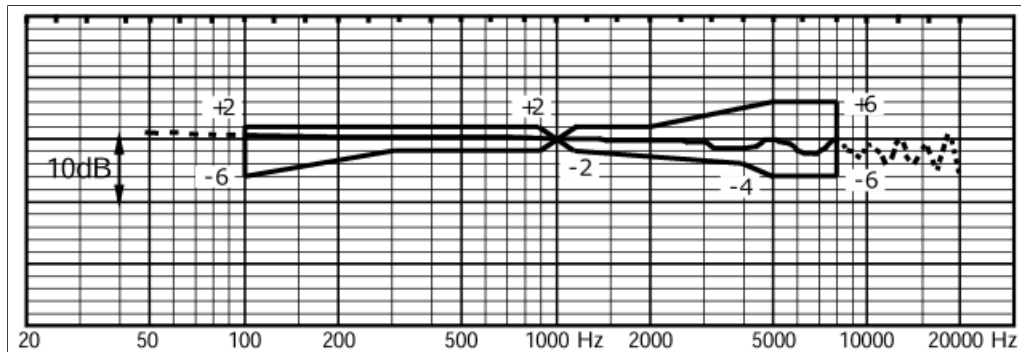


Abbildung 2.6: Frequenzgang des Primo EM158 [2]

Was in den Datenblättern von Mikrofonen auffällt ist der Hinweis der A-Bewertung. Diese Pegelbewertung versucht die objektive Größe Schalldruck und subjektive Größe Lautstärke so miteinander zu vereinen, dass die Hörcharakteristik des menschlichen Ohres, die sonst in ihrem gesamten Umfang nur äußerst aufwendig nachgebildet werden könnte, besser berücksichtigt wird. Angaben zu Pegelgrößen wie dem SNR werden daher immer mit einem zusätzlichen Index oder einer Einheitenergänzung versehen, beispielsweise  $\text{dBA}$  [1].



# 3 Hardware

## 3.1 MEMS Mikrofon ADMP441

### 3.1.1 Eigenschaften MEMS-basierter digitaler Mikrofone

Das ADMP441 von Analog Devices ist ein digitales MEMS Mikrofon. Dies bedeutet, dass die eigentliche Mikrofon-Membran gemeinsam mit einem Vorverstärker, einem Analog-Digital-Wandler (kurz AD-Wandler) sowie einigen anderen Komponenten (digitaler Filter, etc.) auf einem gemeinsamen, meist in CMOS Technik realisierten Chip integriert wird. MEMS Mikrofone bilden dabei in der Regel das Prinzip des Kondensator-Mikrofons nach [9, S. 196].

Ein großer Vorteil, der aus der CMOS-Technik hervorgeht, ist der, dass bekannte und industriell ausgereifte Herstellungsprozesse genutzt werden können, um große Stückzahlen kostengünstig zu produzieren. Dies ermöglicht Mikrofone vielfach mit gleichbleibenden Eigenschaften herzustellen. Dadurch sollen Parameterschwankungen stark eingeschränkt werden können [10].

Eine für den Einsatz in der photoakustischen Spektroskopie wesentlich relevantere Eigenschaft von MEMS Mikrofonen soll die geringere Störempfindlichkeit gegenüber mechanischen Vibrationen im Vergleich zu typischen Kondensator-Mikrofonen sein. Dies wird als Folge der geringeren Oberflächendichte der Membran angegeben [10]. Wenn

$$p_a = \rho \cdot t \cdot a \quad (3.1.1)$$

der äquivalente Druck auf die Membran ist, der durch Vibrationen verursacht wird, dann kann die Vibrationsunterdrückung, das Verhältnis der Membranauslenkungen durch einen Schalldruck  $p$  und durch die Vibrationsbeschleunigungen, durch

$$\frac{x_p}{x_{accel}} = \frac{p}{\ddot{x} \cdot t \cdot \rho} \quad (3.1.2)$$

ausgedrückt werden [9, S. 200].  $\rho$  ist die Materialdichte der Membran,  $t$  die Dicke der Membran und  $a$  die Vibrationsbeschleunigung. Es sollte also von Vorteil sein,  $t \cdot \rho$  möglichst klein zu halten. Mechanische Vibrationen in Form von Körperschall stellen unter anderem

eine große Störquelle bei photoakustischen Messungen dar [5, S. 32f]. Deswegen kann eine erhöhte Vibrationsunterdrückung ein entscheidender Parameter bei der Verbesserung der Messqualität und damit der Nachweisempfindlichkeit sein.

Als weiterer Vorteil von MEMS-Mikrofonen wird angeführt, dass die Empfindlichkeit in Abhängigkeit von der Temperatur nur sehr wenig streuen soll. So sollen typische Streuungen von nur etwa  $0,5\text{dB}$  zu erwarten sein [10]. Bezogen auf ihre Gesamtfläche scheinen MEMS-Mikrofone also konventionellen und gleichgroßen Kondensator-Mikrofonen grundsätzlich bessere Leistungswerte zu besitzen [9, S. 197].

### 3.1.2 Vergleich der Spezifikationen von analogen und digitalen Mikrofonen

Ein wichtiger Parameter von Mikrofonen ist die Empfindlichkeit, auch Sensitivität genannt. Für analoge Mikrofone ist sie das Verhältnis der Ausgangsspannung zum Schalldruckpegel (engl. sound pressure level, SPL). Der Schalldruckpegel ist das logarithmische Verhältnis aus dem Effektivwert des Schalldrucks  $p$  und der Hörschwelle bei  $1\text{kHz}$  als Bezugswert  $p_0 = 20\mu\text{Pa}$  [1, S. 7]

$$L = 20 \cdot \log\left(\frac{p}{p_0}\right) \text{dB SPL} \quad (3.1.3)$$

Die Sensitivität wird typischerweise mit einem Sinuston bei  $1\text{kHz}$  und einem Schalldruckpegel von  $94\text{dB SPL}$  bestimmt, was ca.  $1\text{Pa}$  entspricht, und in der Regel als logarithmische Größe bezogen auf  $1\text{V/Pa}$  angegeben wird [11]:

$$\text{Sensitivity}_{\text{dBV}} = 20 \cdot \log\left(\frac{\text{Sensitivity}_{\text{mV/Pa}}}{1\text{V/Pa}}\right) \text{dBV} \quad (3.1.4)$$

Die Sensitivität von digitalen Mikrofonen unterliegt gewissen Restriktionen. So ist der maximale Schalldruckpegel die entscheidende Größe, da dieser das Full-Scale Wort des AD-Wandlers darstellt. Hier wird das Wort des AD-Wandlers bei einem Schalldruckpegel von  $94\text{dB SPL}$  mit dem Full-Scale Wort in ein logarithmisches Verhältnis gesetzt:

$$\text{Sensitivity}_{\text{dBFS}} = 20 \cdot \log\left(\frac{\text{Sensitivity}_{\%FS}}{(2^n - 1)}\right) \text{dBFS} \quad (3.1.5)$$

Die Pseudoeinheit dBFS berücksichtigt noch einmal den Bezug auf das Full-Scale Wort. Damit ist der maximale Schalldruckpegel die obere Grenze des Aussteuerbereichs eines digitalen Mikrofons. Für ein digitales Mikrofon wie ADMP441, dessen maximaler Schallpegel  $120\text{dB SPL}$  beträgt, muss zwangsläufig gelten, dass die Sensitivität die Differenz aus

dem Referenzschallpegel und dem maximalen Schallpegel ist [12]:

$$Sensitivity_{dBFS} = 94dB SPL - 120dB SPL = -26dBFS \quad (3.1.6)$$

Wegen dieser relativ unterschiedlichen Definitionen, können analoge und digitale Mikrofone nicht direkt anhand der Sensitivität verglichen werden. Weiter ist zu beachten, dass sich die Ausgabe des digitalen Mikrofons bei einem sinusförmigen Eingangssignal auf den Spitzenwert bezieht und nicht wie bei analogen Mikrofonen auf den Effektivwert. Der Effektivwert ist dann  $3dBFS$  kleiner. Allerdings gelten für beide Mikrofontypen die großen der akustischen Domäne, wie der Schalldruckpegel und der SNR [12].

### 3.1.3 Betriebsparameter des ADMP441

Die Abbildung 3.1 zeigt das Blockschaltbild des ADMP441. Der integrierte AD-Wandler arbeitet nach dem Sigma-Delta Prinzip [13, S. 204] mit einer internen Auflösung von 1 Bit und einem Überabstastfaktor von 64 mal der eingestellten Abtastfrequenz. Die minimale und maximale Abtastfrequenz beträgt  $7,8kHz$  beziehungsweise  $49,3kHz$ .

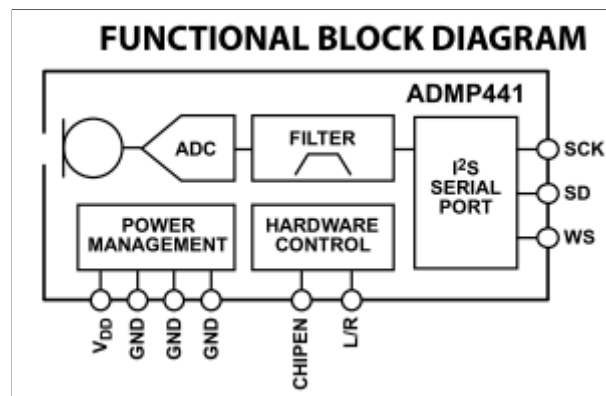


Abbildung 3.1: Blockschaltbild des ADMP441. Quelle: Analog Devices

Durch die hohe Überabstastung verteilt sich die Quantisierungsrauschleistung auf die gesamte Frequenzbandbreite des Wandlers. Der AD-Wandler des ADMP441 quantisiert jeden Abtastwert mit 24 Bit und kodiert sie binär in PCM (Pulse Code Modulation) im Zweierkomplement. In dem Datenblatt des ADMP441 steht allerdings, dass der Effektivwert des Eigenrauschens  $-87dBFS$  beträgt. Damit belegt das Eigenrauschen die unteren 8 bis 9 Bit. Kleinere Werte als  $120dB SPL - 87dBFS = 33dB SPL$  können von dem Mikrofon also nicht mehr erfasst werden und verschwinden im systemeigenen Rauschen.

Durch die 24 Bit Wortbreite wird allerdings der Dynamikumfang deutlich erhöht und der Abstand zwischen zwei Quantisierungsstufen verringert, so dass kleinere Schalländerungen auch am Ausgang wahrgenommen werden können. Um von dem Datenwort  $d$  auf den

Schalldruckpegel zu schließen, kann die folgende Formel genutzt werden [14]

$$L_{SPL} = 120dB_{SPL} - 20\log\left(\frac{d}{(2^{23} - 1)}\right) dB_{SPL} \quad (3.1.7)$$

Neben dem AD-Wandler sind in dem ADMP441 ein Hochpass, zur Reduktion eines Gleichanteils und sehr tiefer Frequenz, und ein digitaler Tiefpass integriert. Der Tiefpass besitzt einen auf die eingestellte Abtastrate  $F_s$  normierten Frequenzgang, der in Abbildung 3.2 zu sehen ist. Seine  $3dB$  Grenzfrequenz liegt bei  $0.423 \cdot F_s$  und im Durchlassbereich besitzt er einen Rippel von  $0,04dB$ . Die minimale Dämpfung im Sperrbereich beträgt  $60dB$ .

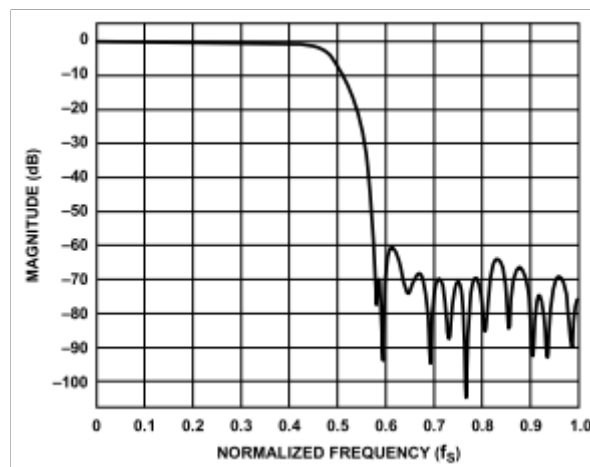


Abbildung 3.2: Frequenzgang des digitalen Tiefpassfilters des ADMP441. Quelle: [14]

Angesteuert wird das ADMP441 über die I<sup>2</sup>S Schnittstelle (Inter-Integrated Circuit Sound Interface), einer von Philips entwickelten, seriellen Schnittstelle zum Austausch von Audiodaten zwischen ICs. I<sup>2</sup>S beschreibt eine Punkt-zu-Punkt Verbindung mit einem Master und einem Slave. Besitzt der Mikrocontroller, an den das Mikrofon angeschlossen werden soll keinen eigenen I<sup>2</sup>S Transceiver, so kann dieser auch über eine eventuell vorhandene SPI Einheit (Serial Peripheral Interface) emuliert werden (siehe [15]).

Die I<sup>2</sup>S Schnittstelle besitzt drei Signalpfade.

- Das Word Select (WS) Signal schaltet bei einem Stereo-Betrieb mit zwei Mikrofonen zwischen dem linken und rechten Kanal um und entspricht damit der Abtastfrequenz.
- Der Serial Clock (SCK) entspricht dem 64-fachen der Abtastfrequenz und ist damit der Eingangstakt des Sigma-Delta AD-Wandlers.
- Der Serial Data (SD) Pfad dient der Übertragung des bitweisen Datenstroms.

Eine Eigenheit des Standard I<sup>2</sup>S Formats ist, dass das MSB eines Datenwortes als erstes übertragen wird und einen SCK Takt nach dem WS Wechsel auf der Leitung ansteht, wie in Abbildung 3.3 zu erkennen ist. Diese Eigenheit muss berücksichtigt werden, falls ein Mikrocontroller ohne I<sup>2</sup>S Transceiver verwendet wird. In dem Fall ist das erste Bit nach dem WS Wechsel auf der SD Leitung das LSB des vorherigen Datenwortes.

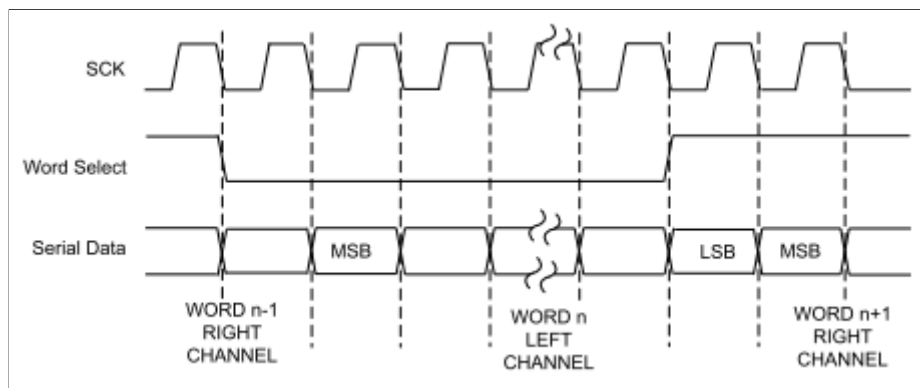


Abbildung 3.3: Datentransfer der I<sup>2</sup>S Schnittstelle. Quelle: [3]

## 3.2 Stellaris EKI-LM3S9B92

Das Stellaris EKI-LM3S9B92 ist ein Evaluationsboard von Texas Instruments. Der auf ihm verbaute LM3S9B92 Mikrocontroller besitzt einen energieeffizienten [16, S. 157] ARM Cortex M3 Kern, der mit einem Takt von bis zu 80 MHz betrieben werden kann. Daneben besitzt er eine große Anzahl an Peripheriegeräten. Besonders sind die Unterstützung von USB 2.0 im Device und Host Modus sowie USB On-the-go (USB OTG), Ethernet und die für das ADMP441 notwendige I<sup>2</sup>S Schnittstelle.

Der ARM Cortex M3 ist ein 32 Bit RISC-Prozessor, der auf der Harvard-Architektur basiert. Die Trennung von Daten- und Befehlsspeicher über separate Bussysteme ermöglicht das parallele Laden und Speichern von Daten und Befehlen. Dadurch entsteht ein Geschwindigkeitsvorteil gegenüber der von-Neumann Architektur, die einen gemeinsamen Bus für Daten- und Befehlsspeicher vorsieht. Dennoch liegen die Daten und Befehle beim Cortex M3 im selben logischen Adressraum. Des weiteren besitzt der Cortex M3 als Hardwareeinheit sowohl einen Multiplizierer (32 x 32 Bit in einem Taktzyklus) als auch einen Dividierer (2 bis 12 Taktzyklen), was ihn auch für Aufgaben der digitalen Signalverarbeitung sehr interessant macht [16, S. 158].

Allerdings besitzt der ARM Cortex M3 keine Floating Point Unit (FPU), wie man der Abbildung 3.4 entnehmen kann. Daher müsste die Rechnung mit Fließkommazahlen durch den

Compiler in Software emuliert werden, wodurch sich der Rechenaufwand und Speicherbedarf erhöht. Stattdessen bietet es sich an, auf dem M3 ein Festkomma-Format für Berechnungen der digitalen Signalverarbeitung zu wählen.

Eine Alternative stellt der ARM Cortex M4 dar, der sowohl eine FPU als auch einen um DSP Instruktionen erweiterten Befehlssatz besitzt sowie SIMD (single instruction, multiple data) Befehle unterstützt. Dies macht ihn noch interessanter für die digitale Signalverarbeitung in eingebetteten System. Eine kleine Gegenüberstellung zwischen dem M3 und M4 hinsichtlich seiner Leistungsfähigkeit in der digitalen Signalverarbeitung findet man unter [17]. Zum Zeitpunkt dieser Arbeit stand allerdings noch kein ARM Cortex M4 Evaluation Board zur Verfügung, das sowohl USB 2.0 als auch I<sup>2</sup>S unterstützt.

Dennoch kann davon ausgegangen werden, dass die Leistungsreserven des ARM Cortex M3 vollkommen ausreichend für die an ihn gestellten Anforderungen sein sollte. Daher und wegen der Kombination der USB 2.0 und I<sup>2</sup>S Schnittstellen, eignet sich das Stellaris EKI-LM3S9B92 Evaluation Board für die Aufgabenstellung dieser Arbeit.

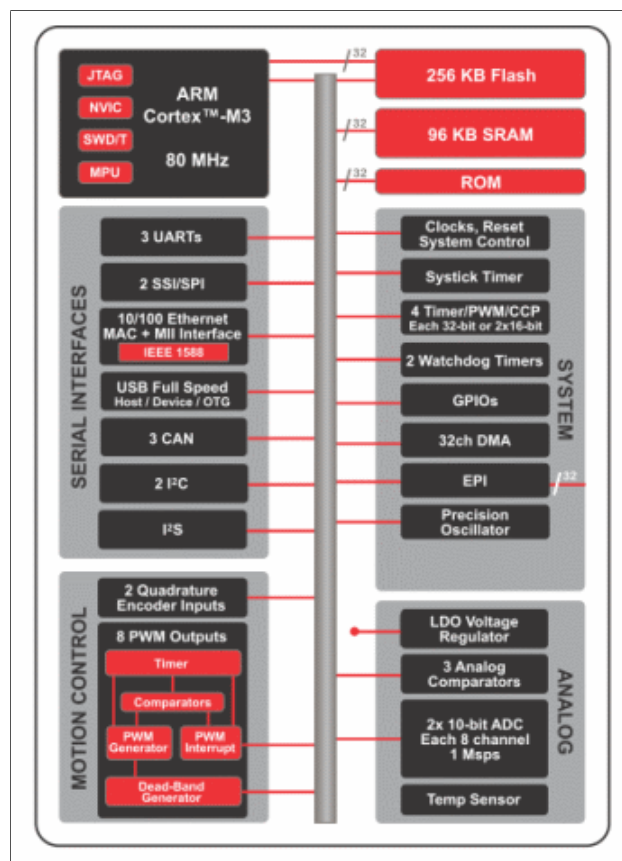


Abbildung 3.4: Überblick über die Komponenten des LM3S9B92. Quelle: Texas Instruments

---

Um die Konfiguration der Hardware leichter, übersichtlicher und wiederverwendbarer zu machen, bietet Texas Instruments das kostenlose Softwarepaket StellarisWare an. In diesem Paket sind neben High Level Gerätetreibern auch Beispielprojekte, sowie hilfreiche Bibliotheken. Zum Beispiel sind eine im Umfang eingeschränkte Implementierung für die Standardausgabefunktion *printf* mit Hilfe der seriellen Ausgabe über die UART Einheit, *UARTprintf*, und ein in seine Anzahl unterstützter Geräteklassen umfangreicher USB Stack enthalten. Die Beispielprojekte enthalten Konfigurationsbeispiele für nahezu alle Peripheriekomponenten der Stellaris Produktlinie und erleichtern den Einstieg in die Programmierung und Inbetriebnahme der Mikrocontroller.

Programmiert wird das Stellaris EKI-LM3S9B92 Evaluation Board mit der Entwicklungsumgebung Code Composer Studio in der Version 5.2.1.00018 mit dem TMS470 Compiler in der Version 4.9.7 und der Stellarisware Version von 2012.

## 4 Signalverarbeitung

In diesem Kapitel sollen die für die photoakustische Spektroskopie ermittelten Signalverarbeitungsalternativen vorgestellt und ihre Einsatzmöglichkeiten untersucht werden.

### 4.1 Phasenempfindlicher Verstärker (Lock-In)

Der in Abschnitt 2.2 genannte Lock-In Verstärker ist ein sehr gutes Werkzeug bei der Messung von Signalen, die stark verrauscht sind. Sein Prinzip beruht auf einer Form der Kreuzkorrelation zwischen dem verrauschten, sinusförmigen Messsignal 2.1.3 und dem Referenzsignal des Chopper-Rades,

$$r(t) = r_0 \cdot \sin(\omega t), \quad (4.1.1)$$

zwischen denen kein zeitlicher Versatz besteht [18, S. 152f]:

$$\begin{aligned} k_{sr}(0) &= s_0(t) \cdot r_0 \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T s(t) \cdot r(t) dt \\ &= s_0(t) \cdot r_0 \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T \left\{ \frac{1}{2} [\cos(\varphi) - \cos(2\omega t + \varphi)] + n(t) \cdot \sin(\omega t) \right\} dt \\ &= s_0(t) \cdot r_0 \cdot \cos(\varphi) \end{aligned} \quad (4.1.2)$$

Durch dieses Vorgehen wird das Messsignal auf die Frequenz 0 Hz herunter moduliert und mit dem Faktor  $r_0$  verstärkt. Das Ergebnis ist idealerweise eine Gleichspannung, die von der Amplitude des Eingangssignals und dessen Phasenbeziehung zum Referenzsignal abhängt. Das Rauschen wird bei einer genügend hohen Integrationsdauer herausgefiltert. Der Lock-In Verstärker stellt damit einen extrem schmalbandigen und über das Referenzsignal abstimmbaren Bandpass Filter dar, der das Signal-Rausch-Verhältnis durch die starke Begrenzung der Bandbreite deutlich verbessern kann. Ein einfaches Blockschaltbild ist in Abbildung 4.1 zu sehen



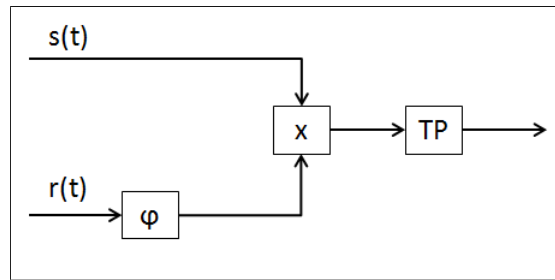


Abbildung 4.1: Funktionsblöcke eines Lock-In Verstärkers.

Da die Phase zwischen dem Mess- und Referenzsignal im ungünstigsten Fall  $90^\circ$  betragen kann, führt dies zu einer Auslöschung des Messsignals. Um dieses Problem zu umgehen, ist ein Phasenschieber notwendig. Die Verwendung eines Phasenschiebers macht die kontinuierliche Bestimmung und Nachregelung des Phasenverhältnisses von Eingangs- und Referenzsignal notwendig. Ein alternativer Ansatz ähnelt einem IQ-Demodulator, der die Beziehung

$$\cos^2(x) + \sin^2(x) = 1 \quad (4.1.3)$$

nutzt. Das Eingangssignal wird dann in zwei separate Lock-In Zweige aufgespalten. Der In-Phase Zweig erzeugt

$$x_i(t) = s_0(t) \cdot r_0 \cdot \cos(\varphi) \quad (4.1.4)$$

und der Quadratur Zweig erzeugt

$$x_q(t) = s_0(t) \cdot r_0 \cdot \sin(\varphi) \quad (4.1.5)$$

wobei  $\varphi$  die Phase zwischen dem Eingangssignal und dem Referenzsignal ist. Anschließend werden beide Zweige quadriert, addiert und abschließend wird die Quadratwurzel aus der Summe gezogen. Dadurch und unter Zuhilfenahme von Gleichung 4.1.3 wird das Ausgangssignal des Lock-In Verstärkers unabhängig von der Phase zwischen dem Eingangssignal und dem Referenzsignal

$$\begin{aligned} k_{sr}(0) &= \sqrt{x_i^2(t) + x_q^2(t)} \\ &= \sqrt{(s_0(t) \cdot r_0 \cdot \cos(\varphi))^2 + (s_0(t) \cdot r_0 \cdot \sin(\varphi))^2} \\ &= s_0(t) \cdot r_0 \cdot \sqrt{\cos^2(\varphi) + \sin^2(\varphi)} \\ &= s_0(t) \cdot r_0 \end{aligned} \quad (4.1.6)$$

Für eine praktische Implementierung eines Lock-In Verstärkers wird entsprechend ein si-

sinusförmiges Signal benötigt und für die Realisierung mit I- und Q-Zweigen ein  $90^\circ$  Phasenschieber. Ein solcher Sinusgenerator lässt sich z.B. mit dem Verfahren der direkten digitalen Synthese (DDS, direct digital synthesis) realisieren. Dabei wird häufig nur ein viertel einer Sinusschwingung in einem ROM gespeichert. Die Adresse im Speicher beschreibt die Phase der Schwingung. Eine volle Sinus- oder Cosinusschwingung lässt sich dann durch entsprechendes Adressieren des ROM realisieren. Über einen Phasenakkumulator, der meist eine deutlich höhere Wortbreite besitzt, als der Adressraum der Sinustabelle, wird die Phase in festgelegten Schrittweiten  $\Delta\varphi$  pro Takt erhöht. Bei gegebenem Takt  $f_C$ , dem Phaseninkrement  $\Delta\varphi$  sowie der Wortbreite  $m$  des ROM, ergibt sich mit

$$f = \frac{\Delta\varphi}{2^m} f_C \quad (4.1.7)$$

die momentan generierte Frequenz[19, S. 861f].

Um eine möglichst stabile Schwingung zu erzeugen, ist es notwendig, dass für den Akkumulator und den Zugriff auf das ROM konstante Laufzeiten gelten. Bei einer Realisierung in Software wird der Generator daher häufig in Assembler implementiert, um unabhängig von Compiler Implementierungen und Optimierungen zu sein. Eine Alternative zur Realisierung in Software ist der Einsatz eines speziellen DDS Bausteins, wie zum Beispiel dem AD9833 von Analog Devices.

## 4.2 Fourier-basierte Spektralanalyse

Das im vorherigen Abschnitt beschriebene Lock-In Verfahren läuft im Zeitbereich ab und setzt ein ebenfalls sinusförmiges Referenzsignal voraus. In der photoakustischen Spektroskopie ist primär die Amplitude des Messsignals von Interesse und weniger die Phasenbeziehung zwischen Mess- und Referenzsignal. Zur Reduktion des Berechnungsaufwands wäre daher ein Verfahren interessant, das ausschließlich die bekannte Chopper-Frequenz zur Bestimmung der Signalstärke voraussetzt, nicht aber ein komplettes Referenzsignal. An dieser Stelle wird der Wechsel von dem Zeit- in den Frequenzbereich interessant. Hier kann das Signal mit bekannten Verfahren der digitalen Signalverarbeitung analysiert werden und die Erzeugung ganzer Zeitsignale entfällt.

Für die Frequenzanalyse eines digitalisierten Zeitsignals wird in der Regel die diskrete Fourier-Transformation (kurz DFT) mit ihrer effizienten Implementierung über die schnelle Fourier-Transformation (FFT, engl. fast fourier-transform, siehe z.B. [4]) verwendet. In der photoakustischen Spektroskopie ist, wie in Abschnitt 2.1 beschrieben, lediglich die Untersuchung einer einzelnen Spektrallinie im Frequenzspektrum des Messsignals von Interesse. Alle weiteren Frequenzkomponenten des Signals stellen in der Regel Rauschen dar. Die Motivation der Signalanalyse im Frequenzbereich durch Fourier-basierende Algorithmen

ist die, dass die DFT eine gute Approximation der Fourier-Reihe für endliche Eingangsssequenzen ist. Die Fourier-Reihe ist in der Lage, periodische Funktionen in einer vollständig äquivalenten Form zum Zeitsignal darzustellen. In der Regel kann aber jedes Messsignal nur für einen endlichen Zeitraum betrachtet werden, so dass, unter Einhaltung des Abtasttheorems [20, S. 297], das Ergebnis der DFT eine gute Schätzung des tatsächlichen Wertes des gesuchten Fourier-Koeffizienten ist. Es kann dann angenommen werden, dass der auf die Länge  $N$  der DFT normierte Wert des  $k$ -ten DFT Wertes und der  $k$ -te Koeffizient der komplexen Fourier-Reihe denselben Informationsgehalt besitzen [4, S. 167]:

$$c_k = \frac{1}{N} X[k] \quad (4.2.1)$$

Über diese Beziehung kann ein Rückschluss auf die Amplitude der Druckänderung innerhalb der Messzelle gezogen werden, die ein Maß für die Konzentration einer Stoffmenge in einem untersuchten Gasmisch ist. In der Praxis, wo nicht alles ideal ist, muss man Gleichung 4.2.1 wohl aber eher als gute Approximation sehen.

Die DFT der Länge  $N$  kann man sich als eine Filterbank vorstellen, die aus  $N$  Bandpässen mit jeweils der Mittenfrequenz

$$f_k = k \cdot \frac{F_s}{N}, \quad k \in \mathbb{Z} \quad (4.2.2)$$

besteht. Abbildung 4.2 verdeutlicht diese Interpretation grafisch.

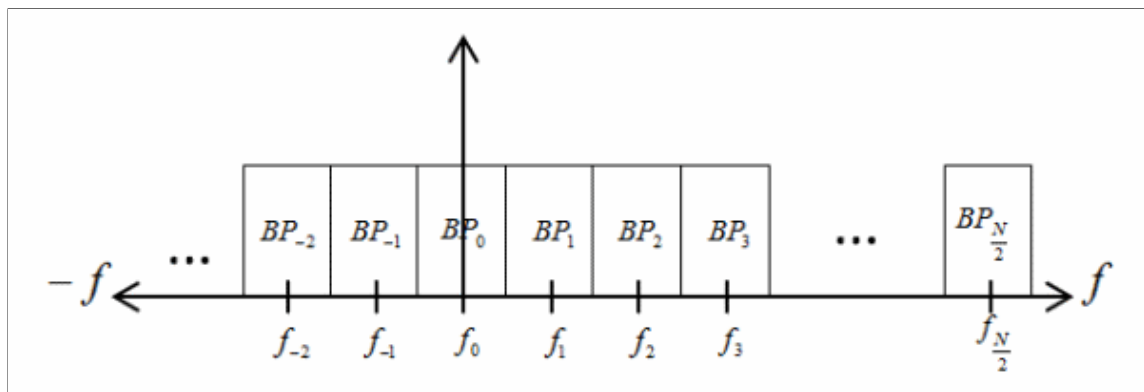


Abbildung 4.2: DFT Spektrum als Filterbank nach [4].

Die Bandbreite der jeweiligen Bandpässe ist von der gewählten Abtastfrequenz  $F_s$  und der Anzahl der berücksichtigten Eingangswerte  $N$  abhängig. Wird die Bandbreite zu groß gewählt und damit die Frequenzauflösung zu gering, kann es dazu kommen, dass Frequenzkomponenten des Eingangssignals den Ausgang eines der Bandpässe anregt, obwohl die eigentliche Frequenz von der Frequenzauflösung der DFT gar nicht abgedeckt wird. Dies ist der so genannte picket fence effect [4, S. 188]. Eine hohe Auflösung erhöht die Mess-

dauer sowie den Speicher- und Rechenaufwand, da im Fall der FFT alle Eingangswerte blockweise verarbeitet und für die Berechnung vorgehalten werden müssen.

Die bei dieser Arbeit verwendete Messzelle besitzt eine gute Resonanz bei ca. 2,7 kHz. Um dieses Signal unter Einhaltung des Abtasttheorems digitalisieren zu können, wird eine Abtastfrequenz

$$F_s > 5,4 \text{ kHz} \quad (4.2.3)$$

benötigt. Die tiefste einstellbare Abtastfrequenz des ADMP441 ist 7,8 kHz. Um eine Auflösung des DFT Spektrums von 1 Hz zu erreichen, müssen 7800 Abtastwerte verarbeitet werden. Jeder Abtastwert wird mit 24 Bit quantisiert und in einer 32 Bit Integervariable gespeichert. Für ein volles DFT Spektrum, das mit dem FFT-Algorithmus berechnet wird, 32 Bit breite Datenspeicher nutzt und folglich 7800 Linien besitzt, wird für die Speicherung der Abtastwerte und der komplexen Rechenergebnisse eine Speichergröße von

$$3 \cdot 7800 \cdot 4 \text{ Byte} = 93,6 \text{ Kilobyte} \quad (4.2.4)$$

benötigt. Das sind bereits 97,5% des SRAM des LM3S9B92 Mikrocontrollers. Dabei wurde angenommen, dass das Ergebnis der FFT komplex ist und der Real- und Imaginärteil jeweils 32 Bit im Speicher belegt. Eine Erhöhung der Modulationsfrequenz für andere Messzellen hat die Erhöhung der Abtastfrequenz zur Folge. Die reine Speicherung der Eingangswerte und der Rechenergebnisse des FFT-Algorithmus kann zur vollständigen Ausschöpfung oder Überschreitung der Speicherressourcen führen.

Für ein DFT Spektrum der Länge  $N$  werden für die Berechnung durch den FFT-Algorithmus nach Cooley und Tuckey  $N \log_2(N)$  Operationen benötigt, wenn  $N$  eine Zweierpotenz ist. Eine Operation besteht dabei aus einer komplexen Multiplikation, die aus zwei reellen Multiplikationen und zwei reellen Additionen besteht, und einer komplexen Addition, die aus zwei reellen Additionen besteht [21, S. 177]. Ist  $N$  keine Zweierpotenz, dann wird die Eingangssequenz durch das Anhängen von Nullen auf die nächste Zweierpotenz verlängert. Dieses Vorgehen wird als Zero Padding bezeichnet [21, S. 178].

Für die oben genannten Zahlenwerte würden sich für das komplette DFT Spektrum der ergänzten Länge  $N = 8192$  ein Aufwand von

$$N \log_2(N) = 106496 \quad (4.2.5)$$

komplexen Operationen ergeben. Dieser Aufwand ist enorm vor dem Hintergrund, dass nur eine einzelne Spektrallinie von Interesse ist. Eine für diesen Zweck viel effizientere Methode zur Bestimmung der gewünschten DFT Spektrallinie ist der Goertzel-Algorithmus, der im nächsten Abschnitt vorgestellt wird.

### 4.3 Goertzel-Algorithmus

Der Goertzel-Algorithmus bietet eine effiziente Möglichkeit der Berechnung der DFT Komponenten im Frequenzspektrum eines Signals, wenn nur wenige Linien von Interesse sind. Mit am häufigsten wird er daher zum Beispiel in der Telefonie beim Mehrfrequenzwahlverfahren (engl. DTMF, dual-tone multi-frequency signaling) eingesetzt [22, S. 620]. Dort wird jedem Tastenfeld eines Telefons ein Signal aus zwei sich überlagernden sinusförmigen Tönen fest zugeteilt. Die Gegenstelle kann anhand der Signalstärke der jeweiligen Frequenzkomponenten auf die gewählte Rufnummer schließen.

Im Unterschied zur Telefonie, bei der sich die im Signal gesuchten Frequenzen unterhalb von 2 kHz befinden und fest vorgegeben sind, wird der Goertzel Algorithmus für den Einsatz in der photoakustischen Spektroskopie immer wieder seine Messfrequenz neu anpassen müssen. Dies setzt voraus, dass die Resonanzfrequenz der Messzelle kontinuierlich und möglichst genau bestimmt werden kann.

Im weiteren Textverlauf sollen die Ausdrücke Goertzel-Algorithmus und Goertzel-Filter synonym verwendet werden.

#### 4.3.1 Herleitung

Nachfolgend soll der Goertzel-Algorithmus nach [22], [20] und [21] hergeleitet werden. Als Grundlage zur Herleitung des Goertzel Algorithmus dient die Definition der diskreten Fourier Transformation

$$X[k] = \sum_{p=0}^{N-1} x[p] W_N^{pk} \quad (4.3.1)$$

mit dem Twiddlefaktor [4, S. 175]

$$W_N = e^{-j\frac{2\pi}{N}} \quad (4.3.2)$$

und der Identität

$$W_N^{-kN} = 1 \quad (4.3.3)$$

Man erhält durch Einsetzen von Gleichung 4.3.3 in Gleichung 4.3.1 und der Zusammenfassung der Exponenten den folgenden Ausdruck für die Bestimmung der k-ten Spektrallinie der Eingangssequenz [21, S. 179]

$$X[k] = W_N^{-kN} \cdot \sum_{p=0}^{N-1} x[p] W_N^{kp} = \sum_{p=0}^{N-1} x[p] W_N^{-k(N-p)}. \quad (4.3.4)$$

Die rechte Seite von Gleichung 4.3.4 ähnelt dabei der diskreten Faltung

$$y[n] = \hat{x}[n] * \hat{h}[n] = \sum_{r=-\infty}^{\infty} \hat{x}[r] \hat{h}[n-r] = \sum_{r=0}^{N-1} x[r] W_N^{-k(n-r)} \quad (4.3.5)$$

wobei  $\hat{x}[n]$  die kausale und endlich Eingangssequenz

$$\hat{x}[n] = \begin{cases} x[n], & n \leq 0 < N-1 \\ 0, & n < 0, N \leq n \end{cases} \quad (4.3.6)$$

und  $\hat{h}[n]$  die kausale aber unendliche Impulsantwort

$$\hat{h}[n] = \begin{cases} W_N^{-kn}, & n \geq 0 \\ 0, & n < 0, \end{cases} \quad (4.3.7)$$

eines Filters ist. Wertet man Gleichung 4.3.5 bei  $n = N$  aus, so kommt man zurück zu Gleichung 4.3.4 und erhält die  $k$ -te Spektrallinie  $X[k]$  der Eingangssequenz. Die Gleichungen 4.3.4 und 4.3.5 lassen sich aber auch als Polynome von  $W_N^{-k}$  interpretieren [20, S. 460],

$$\sum_{n=0}^{N-1} x[n] W_N^{-k(N-n)} = x[0] W_N^{-kN} + x[1] W_N^{-k(N-1)} + x[2] W_N^{-k(N-2)} + \dots \quad (4.3.8)$$

welche sich am Beispiel von Gleichung 4.3.4 über das Horner-Schema zu

$$\sum_{n=0}^{N-1} x[n] W_N^{-k(N-n)} = (\dots (((x[0] W_N^{-k} + x[1]) W_N^{-k} + x[2]) W_N^{-k} + \dots) W_N^{-k} \quad (4.3.9)$$

umstellen lassen. Setzt man  $y[n] = 0$  für  $n < 0$  voraus, so lässt sich der  $n$ -te Ausgangswert in Gleichung 4.3.5 durch

$$y[n] = y[n-1] W_N^{-k} + x[n] \quad (4.3.10)$$

bestimmen. Wendet man auf die Differenzengleichung 4.3.10 die  $z$ -Transformation an, dann erhält man die Übertragungsfunktion eines IIR Filter 1. Ordnung

$$H(z) = \frac{1}{1 - W_N^{-k} z^{-1}}. \quad (4.3.11)$$

mit den Koeffizienten

$$b_0 = 1, \quad a_1 = W_N^{-k}. \quad (4.3.12)$$

Der zugehörige Signalflussplan von Gleichung 4.3.10 ist in Abbildung 4.3 zu sehen. Für reelle Eingangsfolgen sind pro Abtastwert 4 reelle Multiplikationen und 3 reelle Additionen

notwendig.

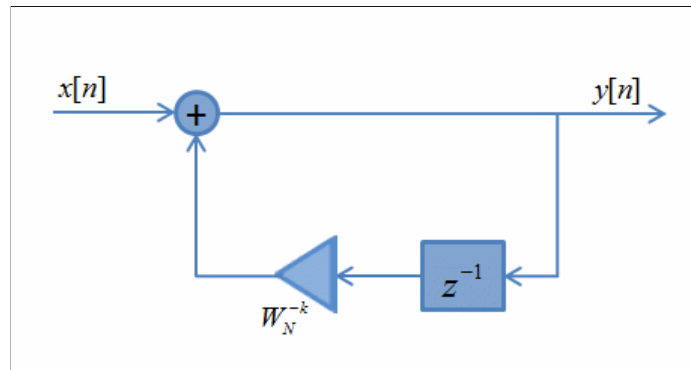


Abbildung 4.3: Signalflussplan des Goertzel-Algorithmus nach Gleichung 4.3.10.

Für die praktische Implementierung dieses Filters wäre es erstrebenswert die komplexe Multiplikation, die für jeden neuen Eingangswert ausgeführt werden muss, zu vermeiden. Die komplexe Multiplikation kann umgangen werden, indem Gleichung 4.3.11 wie folgt modifiziert wird [20, S. 461]

$$\begin{aligned}
 H(z) &= \frac{1}{1 - W_N^{-k}z^{-1}} \cdot \frac{1 - W_N^kz^{-1}}{1 - W_N^kz^{-1}} \\
 &= \frac{1 - W_N^kz^{-1}}{1 - 2\cos(k\frac{2\pi}{N})z^{-1} + z^{-2}} \\
 &= \left( \frac{1}{1 - 2\cos(k\frac{2\pi}{N})z^{-1} + z^{-2}} \right) \cdot (1 - W_N^kz^{-1}) \quad (4.3.13)
 \end{aligned}$$

Durch diese kleine Änderung entstehen zwei neue Filter. Der Ausdruck in der linken Klammer stellt einen IIR Filter 2. Ordnung dar, für den wieder  $y[n] = 0$  für  $n < 0$  gilt. Die Koeffizienten sind

$$b_0 = 1, \quad a_1 = 2\cos(k\frac{2\pi}{N}), \quad a_2 = 1 \quad (4.3.14)$$

Die rechte Klammer stellt einen FIR Filter 1. Ordnung dar, der allerdings nur einmalig für  $n = N$  ausgewertet wird, da die Summe in Gleichung 4.3.5 nur bis  $N - 1$  geht, Gleichung 4.3.4 und 4.3.5 aber nur für  $n = N$  gleich sind. Dessen Koeffizienten sind

$$b_0 = 1, \quad b_1 = W_N^{-k} \quad (4.3.15)$$

Gleichung 4.3.13 nennt sich modifizierter Goertzel-Algorithmus. Aus ihr lassen sich die

Differenzgleichungen

$$v[n] = 2 \cos\left(2\pi \frac{k}{N}\right)v[n-1] - v[n-2] + x[n] \quad (4.3.16)$$

und

$$X[k] = v[N] - v[N-1]W_N^k \quad (4.3.17)$$

ableiten. Abbildung 4.4 zeigt den Signalflussplan des modifizierten Goertzel-Algorithmus. Der gestrichelte Zweig wird nur bei  $n = N$  ausgeführt.

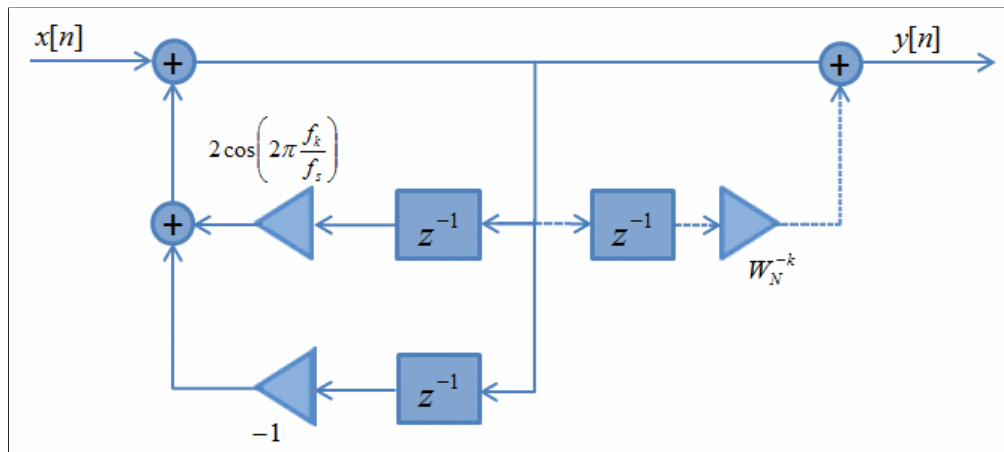


Abbildung 4.4: Signalflussplan des modifizierten Goertzel-Algorithmus nach Gleichung 4.3.16 und 4.3.17.

Um auch die letzte komplexe Multiplikation in 4.3.17 zu umgehen, kann statt dem komplexen Spektralwert die Signalleistung berechnet werden. Sie lässt sich berechnen durch [22, S. 620]

$$P_k = |X[k]|^2 = X[k] \cdot X^*[k] = v^2[N] + v^2[N-1] - 2 \cos\left(2\pi \frac{k}{N}\right)v[N]v[N-1] \quad (4.3.18)$$

Der so gewonnene Wert entspricht dem der positiven Frequenzachse und damit dem einseitigen DFT Spektrum. Möchte man Rückschlüsse auf die Amplitude einer Sinusschwingung ziehen, dann muss auch die Spektralkomponente auf der negativen Frequenzachse mit einbezogen werden. Für die negative Spektralkomponente bleibt der IIR Teil unverändert. Der Unterschied wird nur im FIR Teil durch einen Vorzeichenwechsel des Imaginärteils erkennbar [22].



### 4.3.2 Eigenschaften

Der Abstand zwischen zwei Frequenzen  $\Delta f$  und damit die Auflösung des DFT Spektrums bestimmt sich zu

$$\Delta f = \frac{F_s}{N} = \frac{1}{T_s N}. \quad (4.3.19)$$

Dabei ist  $F_s$  die Abtastfrequenz bzw.  $T_s$  die Abtastperiode und  $N$  die Anzahl der Abtastwerte. Für die  $k$ -te Frequenzkomponente gilt dann

$$f_k = k\Delta f = k \frac{F_s}{N} \quad (4.3.20)$$

Das Argument der Kosinusfunktion in Gleichung 4.3.13 und 4.3.16 lässt sich mit Gleichung 4.3.20 zu

$$\varphi = \frac{k}{N} = \frac{f_k}{F_s} \quad (4.3.21)$$

umformulieren. Damit ist es nur noch von der Abtastfrequenz und der in der Eingangssequenz gesuchten Spektralkomponente abhängig. Anders, als bei der FFT, darf  $\varphi$  reell sein [21, S. 182].

Aus Gleichung 4.3.19 wird schnell ersichtlich, dass die Frequenzauflösung und damit die Genauigkeit der Messung mit der Anzahl der Abtastwerte  $N$  linear steigt. Andererseits bedeutet die Erhöhung der berücksichtigten Messwerte auch eine Verlängerung der Messdauer. Es wird also immer auf einen Kompromiss aus Messdauer und Messgenauigkeit hinauslaufen. Gegenüber der FFT müssen aber, wie bereits am Anfang des Abschnitts angemerkt, nicht alle  $N$  Abtastwerte zwischengespeichert werden. Damit wird die Anzahl der berücksichtigten Abtastwerte im Allgemeinen nicht von der vorhandenen Speicherkapazität beschränkt.

Man kann den Goertzel Algorithmus auch als abstimmbaren Bandpassfilter ansehen, dessen Bandbreite reziprok zur Anzahl der Abtastpunkte ist

$$B \sim \frac{1}{N}. \quad (4.3.22)$$

Kommt bei der Verarbeitung der Abtastwerte keine spezielle Fensterfunktion zum Einsatz, wird zwangsläufig das Rechteck Fenster verwendet, da immer nur  $N$  Abtastwerte für einen Ausgangswert berücksichtigt werden. Somit wird bei der Berechnung nur ein Ausschnitt des Eingangssignals betrachtet. Die Multiplikation im Zeitbereich hat eine Faltung im Frequenzbereich zur Folge. Dies gilt auch für diskrete Signale. Das kausale, zeit-diskrete Recht-

eck Fenster der Länge  $m$

$$w_m[n] = \begin{cases} 1, & \text{für } 0 \leq n \leq m-1, n, m \in \mathbb{N} \\ 0, & \text{sonst} \end{cases} \quad (4.3.23)$$

besitzt über seine zeit-diskrete Fourier-Transformation, also die Fourier-Transformierte für abgetastete Signale, die Darstellung

$$W_m(e^{j\omega}) = \sum_{n=-\infty}^{\infty} w[n] \cdot e^{-j\omega n} = \sum_{n=0}^{m-1} e^{-j\omega n} \quad (4.3.24)$$

Über die geometrische Summenformel und etwas Umformung gelang man zu

$$W_m(e^{j\omega}) = \frac{1 - e^{-j\omega m}}{1 - e^{-j\omega}} = e^{-j\frac{\omega(m-1)}{2}} \cdot \frac{\sin(\frac{\omega m}{2})}{\sin(\frac{\omega}{2})} \quad (4.3.25)$$

Der Betrag von 4.3.25 wird als Dirichlet-Kern bezeichnet und ist quasi das zeit-diskrete Äquivalent zur zeit-kontinuierlichen Si-Funktion [23, S. 111]

$$D_m(x) = \frac{\sin(\frac{\omega m}{2})}{m \cdot \sin(\frac{\omega}{2})} \quad (4.3.26)$$

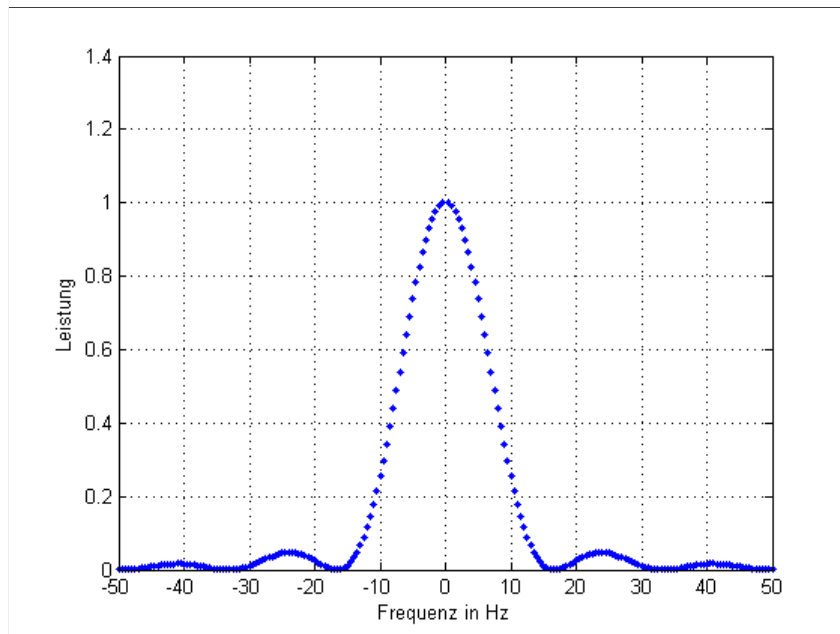


Abbildung 4.5: Leistungsfrequenzgang von 4.3.26. Die Abtastfrequenz ist 100 Hz

Er bestimmt maßgeblich die Bandbreite des Goertzel Filters. Wie bei der Si-Funktion im zeit-kontinuierlichen Fall, bestimmt auch der Dirichlet-Kern über das Zeit-Bandbreite Produkt im diskreten Spektrum die Frequenzauflösung. Ein großes Fenster im Zeitbereich mit vielen Abtastwerten führt zu einem schmalen Durchlassbereich und damit zu einer starken Frequenzselektivität (vgl. [21, S. 183]).

### 4.3.3 Stabilität

Aus Gleichung 4.3.13 lässt sich ablesen, dass der Goertzel Filter sich aus einem IIR Filter 2. Ordnung und einem FIR Filter 1. Ordnung zusammensetzt. Kritisch für die Stabilität ist der IIR Teil, dessen Polstellen auf dem Einheitskreis in der komplexen Ebene liegen. Das System ist also bedingt stabil. Durch die endliche Darstellungsgenauigkeit, speziell bei Festkomma Systemen, kommt es durch Rundungsfehler bei der Quantisierung der Filterkoeffizienten zu Verschiebungen der Pol- und Nullstellen. Dies kann zum einen zu einem grundsätzlich anderen Systemverhalten führen und zum anderen kann es das ohnehin grenz-stabile System vollkommen instabil werden lassen.

Da die internen Speicher des Goertzel Filters nach  $N$  Abtastwerten wieder zurückgesetzt werden, wird die Gefahr einer permanenten, unkontrollierten Schwingung am Ausgang vermindert. Allerdings beeinflusst die eventuelle Instabilität des Filters das momentan berechnete Ausgangssignal. Um dies zu verhindern, wird an die Genauigkeit der Darstellung der Filterkoeffizienten hohe Ansprüche gestellt.

Ein weiteres Problem, das zu unerwünschten Ergebnissen führt, sind Überläufe oder Unterläufe in Folge der zu geringen Wortbreite zur Darstellung der Berechnungsergebnisse und Zwischenschritte. Wie aus dem vorherigen Abschnitt hervorgegangen ist, sind für schmalbandige Messungen die Verarbeitung vieler Abtastwerte notwendig. Dadurch wird die Gefahr von Über- und Unterläufen weiter verstärkt. Eine Gegenmaßnahme wäre die geeignete Skalierung jedes Abtastwertes. Dies kann aber unter Umständen den Verlust von Informationen und damit weitere Ungenauigkeiten nach sich ziehen.

# 5 Implementierung der Firmware

In diesem Kapitel wird die Konfiguration und Inbetriebnahme des LM3S9b92 Evaluation Boards und die Implementierung der Signalverarbeitung erläutert. Vorher werden die genutzten Hardware Blöcke mit ihrer Konfiguration und bestimmten Eigenheiten besprochen.

## 5.1 Konfiguration der Hardware

### 5.1.1 Systemtakt

Das Stellaris LM3S9B92 Evaluation Board kann mit einem maximalen Kerntakt von 80 MHz betrieben werden. Für eine schnellstmögliche Verarbeitungsgeschwindigkeit wird der Kerntakt deswegen auch auf 80 MHz eingestellt. Damit besitzt der Mikrocontroller genug Leistungsreserven für die Signalverarbeitung und Ansteuerung des digitalen MEMS Mikrofons.

Um diesen hohen Systemtakt nutzen zu können, muss der Takt aus der PLL abgeleitet und mit der Funktion SysCtlClockSet() eingestellt werden.

Listing 5.1: Konfiguration des Systemtakts

```
1 /**
2  * SYSCTL_SYSDIV_2_5  -> 80 MHz Systemtakt
3  * SYSCTL_USE_PLL    -> PLL nutzen, damit I2S Master Clock läuft
4  * SYSCTL_OSC_MAIN   -> on-Chip Oszillator als PLL Quelle nutzen
5  * SYSCTL_XTAL_16MHZ -> 16 MHz Oszillator auswählen
6  */
7 SysCtlClockSet(SYSCTL_SYSDIV_2_5 | SYSCTL_USE_PLL | SYSCTL_OSC_MAIN
8                | SYSCTL_XTAL_16MHZ);
```

### 5.1.2 UART

Zur Ausgabe von Messwerten und teilweise zum Debuggen der in Abschnitt 5.2 beschriebenen Firmware per *printf*-Statusausgaben, wird die UART Einheit des Stellaris LM3S9B92 zur Übertragung über die serielle Schnittstelle benötigt. Die Konfiguration der UART Einheit erfolgt dabei über die Initialisierungsroutine der in Abschnitt 3.2 angesprochenen *printf*-Implementierung *UARTprintf*. Um diese Funktion benutzen zu können, muss die Datei *uartstdio.c*, die den Quelltext der Funktionsroutinen enthält, in das aktuelle Code Composer Studio Projekt aus dem Stellarisware Ordner importiert werden. Zur Initialisierung sind die folgenden Codezeilen notwendig

Listing 5.2: Konfiguration des Systemtakts

```
1 // Port A aktivieren
2 SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);

4 // Die Pins A0 und A1 als UART Pins konfigurieren
5 GPIOPinConfigure(GPIO_PA0_U0RX);
6 GPIOPinConfigure(GPIO_PA1_U0TX);

8 // Die alternative Funktion der Pins für UART auswählen
9 GPIOPinTypeUART(GPIO_PORTA_BASE, GPIO_PIN_0 | GPIO_PIN_1);

11 // Initialisierung der UART Schnittstelle
12 UARTStdioInit(0);
```

Anschließend kann über den Hardware Debugger des Stellaris LM3S9B92 mit der Funktionsroutine *UARTprintf* ein Textstring an ein Terminalprogramm, wie beispielsweise Putty, gesendet werden. Die Funktionsroutine *UARTStdioInit(0)* in Listing 5.2 konfiguriert die UART Einheit mit der internen Nummer null vollständig. Eingestellt wird standardmäßig eine Bitrate von 115200 Bit/s, 8 Datenbits, kein Paritätsbit und ein Stopbit.

### 5.1.3 I<sup>2</sup>S

Die I<sup>2</sup>S Schnittstelle benötigt mehrere Taktquellen für die verschiedenen Ein- und Ausgangssignale. Zum einen für das Word Select (WS) Signal, das zwischen dem linken und rechten Kanal einer Stereobeschaltung umschaltet und das den Zeitrahmen für einen Abtastwert festlegt. Der WS Takt entspricht damit der eingestellten Abtastrate. In der Monobeschaltung wird das Signal des zweiten Kanals verworfen. Zum anderen wird ein Serial Clock (SCK) Takt benötigt, der der Beziehung  $SCK = 64 \cdot WS$  genügt.

Der LM3S9B92 wird als RX Master betrieben. Das heißt, dass die I<sup>2</sup>S Schnittstelle des Mikrocontrollers als Datenempfänger und Takterzeuger fungiert. Daher werden die beiden benötigten Takte von Master Clock (MCLK) abgeleitet. Der Master Clock wird über die

Funktion `SysCtlI2SMClkSet` in Listing 5.3 konfiguriert. Die Null als erster Funktionsparameter gibt an, dass der Master Clock über die PLL erzeugt wird. Diese muss dafür vorher aktiviert worden sein, was durch die Einstellungen im vorherigen Abschnitt erfolgt ist.

Listing 5.3: Konfiguration des I<sup>2</sup>S Master Clock

```
1 /**
2 * Den Master Clock (MCLK) Takt festlegen
3 * 8kHz * 32 bits per sample * 2 samples per frame * 4 mclks per sclk
4 * = 256*fs
5 */
6 ulMclkRate = SysCtlI2SMClkSet(0, 8000 * 32 * 2 * 4);
```

Die I<sup>2</sup>S Einheit des LM3S9b92 besitzt einen Eingangs-FIFO, der 8x32 Bit groß ist. Damit können in ihm acht 24 Bit Monosamples oder acht 16 Bit Stereopaare gepuffert werden. Über die Funktion `I2SRxFIFOLimitSet()` kann der FIFO Level eingestellt werden, bei dem ein Service Request ausgelöst wird. Dies kann entweder zu einem Interrupt Request oder zu einem DMA Request führen. Der FIFO Level ist ein Wert zwischen 0 und 16, wobei ein 24 Bit Monosample zweifach zählt.

Der FIFO Level sollte so eingestellt werden, dass der Datenpuffer im Speicher einem Vielfachen des FIFO Levels entspricht. Entspricht der Datenpuffer zum Beispiel einer Größe von 48, dann sollte der FIFO Level auf 16 eingestellt werden. Dadurch wird zum einen nur ein Service Request für den vollen FIFO ausgelöst, weil nicht mehr als acht Monosamples in den FIFO passen. Zum anderen entspricht dann die Größe des Puffers einem Vielfachen der Anzahl der Samples im FIFO. Außerdem wird so möglichst selten ein Service Request ausgelöst, was den Betrieb effizienter macht. Eine mögliche Gefahr besteht darin, dass der Service Request nicht schnell genug bedient werden kann und damit Werte im FIFO überschrieben werden. Im bisherigen Betrieb ist dieses Problem aber nicht in Erscheinung getreten.

Ein falsch eingestellter FIFO Level, für einen Datenpuffer der Größe 48, hat bei der Datenübertragung per USB (siehe Abschnitt 5.2.1) beispielsweise zu massiven Störungen des Signals geführt. Eine Korrektur des FIFO Levels auf den Wert 16 hat das Problem beseitigt.

### 5.1.4 uDMA

Für eine effiziente Übertragung der Audiodaten aus dem I<sup>2</sup>S FIFO in den RAM, um die Daten dort für die Signalverarbeitung bereit zu stellen, wird der in den Mikrocontroller integrierte uDMA Controller (Micro Direct Memory Access Controller) verwendet. Er entlastet den Prozessorkern von den weniger anspruchsvollen Aufgaben des reinen Datentransfers, so dass mehr Rechenkapazität für die Signalverarbeitung zur Verfügung steht.

Der uDMA Controller wird im Ping Pong Modus betrieben. Dieser Modus setzt mindestens zwei Puffer voraus. Während in den einen die Daten des I<sup>2</sup>S FIFO geschrieben werden, können die Daten des anderen gelesen und weiter verarbeitet werden. Für die Konfiguration des Ping Pong Modus sind zwei Strukturen im Speicher notwendig. Die eine legt die Parameter des primären Puffers fest, die andere des alternativen. Wenn der uDMA Kanal der I<sup>2</sup>S Schnittstelle aktiviert ist, wird durch den Service Request des I<sup>2</sup>S FIFO neben einem I<sup>2</sup>S Interrupt ein DMA Request ausgelöst. Der uDMA liest dann den I<sup>2</sup>S FIFO aus und speichert die Daten in dem momentan aktiven Puffer ("ping"), bis dieser voll ist. In diesem Fall wird die aktuelle Pufferstruktur deaktiviert und der uDMA Controller wechselt automatisch in die zweite Pufferstruktur und schreibt die neuen Daten in den nächsten Puffer ("pong").

Für einen kontinuierlichen Datenstrom wird in dem Interrupt Handler der I<sup>2</sup>S Schnittstelle überprüft, ob eine der beiden Pufferstrukturen angehalten wurde. Ist dies der Fall, dann wird über die Funktion uDMAChannelTransferSet() ein neuer Transfer konfiguriert. An dieser Stelle kann beispielsweise auch ein neuer Puffer im Speicher adressiert werden. So wäre es möglich einen Ringpuffer zu nutzen.

### 5.1.5 Timer

Der Goertzel Algorithmus benötigt die momentane Referenzfrequenz, um das photoakustische Signal richtig zu detektieren. Der Regler des Chopper-Rades erzeugt ein Rechtecksignal mit der Periode der momentanen Referenzfrequenz. Die Frequenz kann über dieses Signal durch den Input Capture Eingang von einem der General Purpose Timer gemessen werden.

Der LM3S9B92 bietet vier frei programmierbare Timer Blöcke, die jeweils entweder als ein einzelner 32 Bit oder als zwei 16 Bit Timer betrieben werden können. Bezeichnet werden die 16 Bit Timer als TimerxA und TimerxB. Zur Messung des Referenzsignals wird der Timer0A Block verwendet. Für die Frequenzmessung wird das Referenzsignal an den CCP0 (Capture/Compare) Pin angeschlossen.

Zur Messung einer Frequenz mit einem Mikrocontroller gibt es verschiedene Möglichkeiten. Einige Verfahren eignen sich besser für langsame Signaländerungen, andere eher für schnellere. Der begrenzende Faktor ist dabei der Timer Takt, der in der Regel von dem Systemtakt abgeleitet wird und die obere Grenze des messbaren Eingangssignals festlegt. In diesem Fall ist der Systemtakt und damit der Timer Takt mit 80 MHz deutlich größer, als die Frequenz des Eingangssignals, welches sich in dieser Anwendung zwischen 2 und 3 kHz bewegt.

Hier wird der Timer0A im Input Edge-Time Modus betrieben. Bei einer steigenden Flanke wird der momentane Zählerstand in das GPTMT0R Register des Timers gesichert und ein Interrupt Request ausgelöst. In dem Interrupt Handler des Timer0A wird der gesicherte Zählerstand und der Zählerstand der vorherigen positiven Flanke in einer Variable gespeichert.

Der Timer wird mit einem Startwert geladen und zählt bei jedem Timertakt einen Schritt herunter. Beim Erreichen des Wertes Null wird automatisch der Startwert neu geladen, so dass der Timer in einer Dauerschleife läuft. Der Wert der gemessenen Frequenz ergibt sich aus der Timerfrequenz dividiert durch die Differenz des Timerstandes bei der ersten und der zweiten Flanke

$$f = \frac{f_{\text{Takt}}}{\text{Timerstand1} - \text{Timerstand2}} \quad (5.1.1)$$

Entgegen den Informationen in dem Datenblatt des LM3S9B92 [3], kann der Takt des Timers nicht über einen Vorteiler heruntergesetzt werden. Anfängliche Messungen mit diesem Verfahren haben starke Schwankungen innerhalb weniger Perioden des Referenzsignals auf Grund der hohen Geschwindigkeit des Timers ergeben. Um für die spätere Signalverarbeitung sinnvolle Messwerte der Referenzfrequenz auszugeben, wird über 64 Perioden gemittelt. So bleibt die gemessene Frequenz über sinnvolle Zeiten hin konstant. Über diesen Messweg sind Frequenzmessungen auf eine Genauigkeit von 1 Hz möglich.

## 5.2 Firmware

Innerhalb der Bearbeitung dieser Aufgabenstellung sind zwei Softwareprojekte für den LM3S9B92 entstanden, die in den nachfolgenden Unterabschnitten beschrieben werden sollen.

### 5.2.1 USB Mikrofon Firmware

Um ein Gefühl für das zu erwartende Messsignal zu bekommen und zu überprüfen, ob die Ansteuerung des ADMP441 korrekt funktioniert, war es wichtig das Mikrofonsignal für weitere Untersuchungen aus dem Mikrocontroller herauszubekommen. Ideal wäre ein Übertragungsweg, der das Mikrofonsignal gleich in einem für die Analyse und Erprobung der digitalen Signalverarbeitung, zum Beispiel in Matlab, sinnvollen Format liefert. Da Texas Instruments, wie in Kapitel 3.2 bereits angesprochen, in seinem Softwarepaket StellarisWare einen USB Stack mitliefert, der die USB Audio Device Class unterstützt, fiel die Entscheidung zugunsten der auf dem LM3S9B92 zur Verfügung stehenden USB 2.0 Schnittstelle. Die Alternative wäre die vorhandene Ethernet Schnittstelle gewesen.

Das Stellarisware Paket liefert ein Beispielprojekt, das die USB Audio Device Class in Verbindung mit der I2S Schnittstelle nutzt. Allerdings sieht dieses Beispiel die Übertragung von Audiodaten von einem Host PC zu dem Mikrocontroller vor, um sie dann über die I2S Schnittstelle an einen Lautsprecher zu senden. Für die Übertragung der Mikrofondaten an



den Host PC musste daher der USB Stack umgeschrieben und ergänzt werden. Die wichtigsten Einstellungen für die Übertragung über die USB Schnittstelle werden in den Deskriptoren vorgenommen. Dort wird unter anderem festgelegt, dass das Gerät die USB Audio Device Class nutzt, welche Abtastfrequenz verwendet wird und wie groß die Bitbreite eines Abtastwerts ist. Eine ausführliche Übersicht über die Konfiguration der Deskriptoren findet sich unter [24, Appendix B].

Die USB Mikrofon Firmware konnte bisher nur in Verbindung mit einem PC mit Windows XP verwendet werden. Mit einem USB Protokoll Logger wurde untersucht, warum das USB Mikrofon nicht unter Windows 7 zum laufen gebracht werden konnte. Eine abschließende Fehlerursache konnte nicht ausgemacht werden. Allerdings entstand das Problem bereits bei der Enumerierung, also dem Prozess der Bekanntmachung von USB Host und USB Device. Das USB Mikrofon tauchte daraufhin auch nicht in der Liste der Erkannten Geräte im Gerätemanager des Systems auf. Der Test wurde auf zwei verschiedenen Computern unternommen und scheiterte immer an der selben Stelle.

## 5.2.2 Goertzel-Algorithmus Firmware

Abbildung 5.1 zeigt den Programmflussplan der entwickelten Software zur Implementierung des Goertzel Algorithmus. Bei der Inbetriebnahme des Evaluation Boards werden als erstes die genutzten Peripherieeinheiten konfiguriert und aktiviert, wie es in Abschnitt 5.1 beschrieben wurde. Anschließend geht das Programm in einen endlichen Zustandsautomaten mit drei Zuständen über.

In dem Initialzustand *FreqMeasure* wird als erstes die Messung der Referenzfrequenz vorgenommen, da diese für den Goertzel-Algorithmus zwingend notwendig ist. Außerdem wird in diesem Zustand der zweite Filterkoeffizient des IIR Filters aus Gleichung 4.3.13 berechnet. Im else-Zweig von Listing 5.4 erkennt man, dass dieser Zustand beibehalten wird, solange kein Referenzsignal anliegt. Die gesamte Firmware verbleibt dann in diesem Zustand und es wird keine Ausgabe erzeugt.

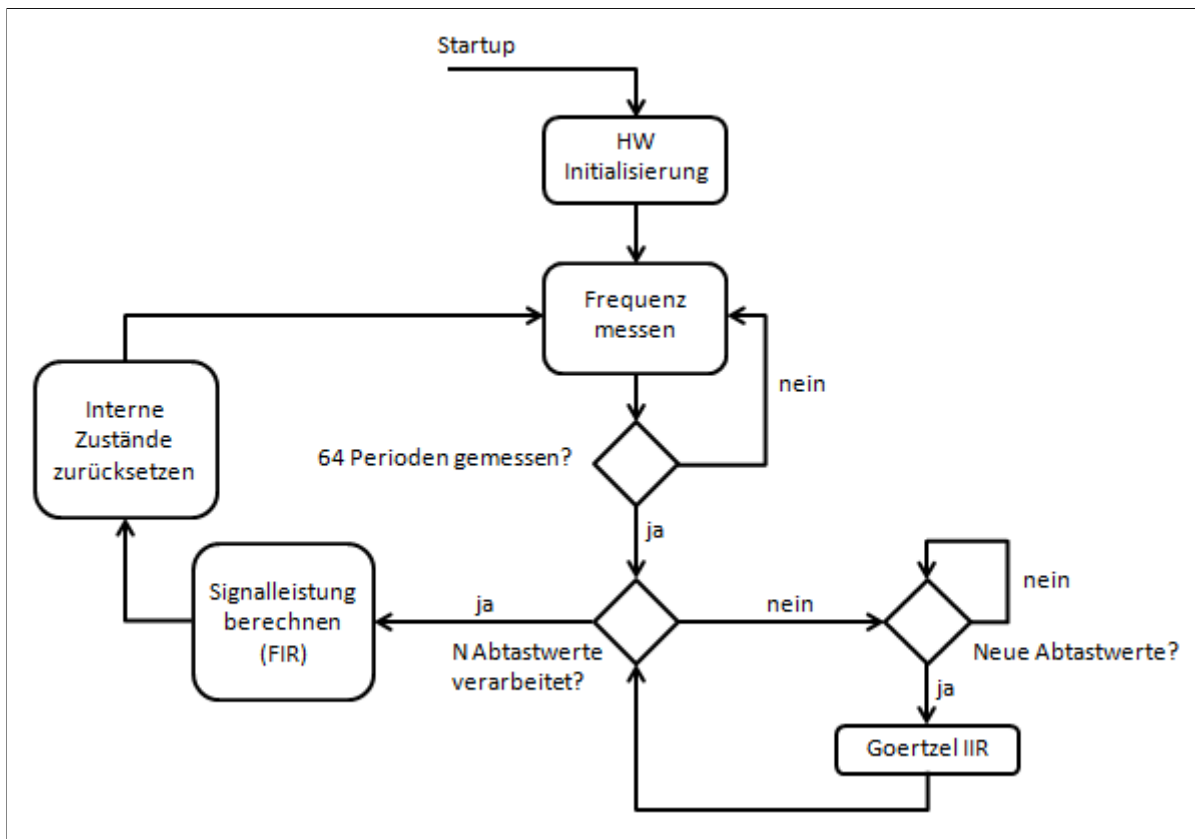


Abbildung 5.1: Programmflussplan der Goertzel-Algorithmus Firmware

Listing 5.4: Referenzfrequenz messen und IIR Koeffizienten berechnen

```

1 case FreqMeasure :
2
3   if (timerEN == 1)
4   {
5     // Zählerstände wurden in Timer ISR ausgelesen
6
7     diffs[diffIndex] = counter1 - counter2;
8     diffIndex++;
9     timerEN = 0;
10
11    if (diffIndex == 63)
12    {
13      for (i = 0; i < 64; i++)
14      {
15        ticks += diffs[i];
16      }
17      ticks = ticks >> 6;
18      measuredFreq = SysCtlClockGet() / ticks;
  
```

```

20 // IIR Koeffizienten neu berechnen
21 varcos = cos(2*PI*((float)measuredFreq/8000.0f));

23 diffIndex = 0;

25 // Zustandswechsel auslösen
26 Zustand = Goertzel;
27 }
28 }
29 else
30 {
31 // Wenn kein Referenzsignal anliegt weiter
32 // in diesem Zustand bleiben
33 Zustand = FreqMeasure;
34 }
35 break;

```

Ist die Frequenzmessung abgeschlossen, wird ein Zustandsübergang ausgelöst. Der Folgezustand *IIR* enthält die Operationen des rekursiven Teils des Goertzel-Algorithmus aus Gleichung 4.3.16. Die Ausführung des Programmcodes ist an zwei Bedingungen gebunden. Zum einen läuft dieser Teil nur so lange, bis die im Quellcode definierte Anzahl *N* an Abtastwerten verarbeitet wurde. Zum anderen wird ein Flag ausgewertet, welches in dem Interrupthandler der I<sup>2</sup>S Einheit gesetzt wird. Dieses Flag gibt an, ob ein neuer Datenpuffer zur Verarbeitung bereit steht. Werden beide Bedingungen erfüllt, dann wird der Code in Listing 5.5 ausgeführt.

Listing 5.5: IIR Filter des Goertzel-Algorithmus

```

1 case IIR :

3   if (counter < N)
4   {
5     if (bufferVoll == 1)
6     {
7       bufferVoll = 0;

9       //Algorithmus durchführen
10      for (i = 0; i < BUFFER_SIZE && counter < N; i++)
11      {

13         // v[n] = 2 * cos(2*pi*k/N) * v[n-1] - v[n-2] + x[n]
14         temp = (float) I2SPuffer[oldIndex][i];
15         temp /= 2147483647.0f;
16         v[0] = (varcos*v[1]) - v[2] + temp;

18         v[2] = v[1];
19         v[1] = v[0];

21         // Samplezähler inkrementieren

```

```
22     counter++;
23     }
24     }
25     }
26     else
27     {
28         // FIR Filter Teil
29         Zustand = FIR;
30     }
31     break;
```

Der Wert der Variable *oldIndex* wird im Interrupthandler der I<sup>2</sup>S Einheit bestimmt. Die Zeilen 14 und 15 in Listing 5.5 sind notwendig, da die Verarbeitung in Fließkommadarstellung durchgeführt werden soll, die 24 Bit breiten Abtastwerte allerdings als 32 Bit Integer gespeichert sind. Da die Abtastwerte die obersten 24 Bit einnehmen, kann man sie auch als Festkommazahl im 1.31 Format interpretieren, wobei die letzten 8 Bit mit Nullen besetzt sind. Zunächst wird der Integerwert nach *float* gecastet und anschließend durch  $(2^{31} - 1) = 2147483647$  dividiert.

Sind N Abtastwerte verarbeitet worden, dann wird ein erneuter Zustandswechsel ausgelöst, der in den Zustand *FIR* führt. In diesem wird gemäß Gleichung 4.3.18 das Betragsquadrat des berechneten Spektralwertes bestimmt.

Listing 5.6: FIR Filter des Goertzel-Algorithmus

```
1 case FIR:
2     power = (v[1]*v[1] + v[2]*v[2] - 2*varcos*v[1]*v[2]);
3     mag = sqrt(power);
4     ergebnis = (long) mag*1000;

6     UARTprintf("%d / %d \n\n", measuredFreq, ergebnis);

8     // Interne Zustände zurücksetzen
9     counter = 0;
10    v[0] = 0.0f;
11    v[1] = 0.0f;
12    v[2] = 0.0f;

14    Zustand = FreqMeasure;
15    break;
```

Um die Amplitudenspektren der Messungen mit dem MEMS- und dem Elektret-Mikrofon vergleichen zu können, wird in dem Zustand *FIR* in Listing 5.6 die Quadratwurzel des Ergebnisses aus Gleichung 4.3.18 bestimmt. Nach der Ausgabe des Filterergebnisses werden die internen Speicher wieder zurück gesetzt und es wird ein Wechsel in den Zustand *FreqMeasure*, womit der in diesem Abschnitt beschriebene Programmablauf erneut beginnt.

Die Ausgabe der Filterergebnisse erfolgt mit der in Abschnitt 3.2 beschriebenen Implementierung der Standardausgabefunktion *printf*, *UARTprintf*. Diese ist im Funktionsumfang dahingehend reduziert, dass die Ausgabe von Fließkommazahlen nicht unterstützt wird. Werte, die als Datentyp *float* gespeichert werden und über diese Funktion ausgegeben werden sollen, werden als Integer behandelt und somit nach dem ganzzahligen Anteil abgeschnitten. Um dennoch Ergebnisse auf drei Nachkommastellen ausgeben zu können, wird daher das Ergebnis aus Zeile 3 in Listing 5.6 mit dem Faktor 1000 multipliziert. Dies muss bei der Auswertung der Ergebnisse berücksichtigt werden, um diese richtig interpretieren zu können.

# 6 MEMS-basierte Photoakustik

## 6.1 Experimenteller Aufbau

In der nachfolgenden Tabelle werden alle Geräte mit ihren Parametern aufgelistet, die neben dem Stellaris LM3S9B92 und dem ADMP441 für die Messungen verwendet wurden.

Tabelle 6.1: Geräteliste

Beschreibung	Parameter	Wert
IPG Photonics Fiber Laser YLR-10-1064-LP-SF	Case Temperatur	33,4°C
Scitec Optical Chopper		
Lockhead Martin Aculight 2400 OPO Modul	Crystal Temperature	50,0°C
	Block Temperature	36,0°C
	Speed Source Tune	0,005VV
PAS-Tech Mikrofonvorverstärker PMV 201	Analoge Filter	aus
	Eingangsspannungsbereich	1,25 mV
Signal Recovery 7265 DSP Lock-In Amplifier	AC Gain	0 dB
	Input Limit	3V
	Sensitivity	100 mV
	TC	1 s
Vacuubrand DVR5 (Druckmessgerät)		
Ophir Nova 2 Laser Power Meter		
Bristol 721 Laser Spectrum Analyzer (Wavemeter)		
Primo EM 158* Ultra High sensitivity Electret Condenser Microphone		
Creative Sound Blaster X-Fi Surround 5.1 Pro USB		

Zu dem Mikrofonvorverstärker existiert keine Dokumentation, weswegen dieser Teil der Messkette prinzipiell als Blackbox betrachtet werden muss.

Als Prüfgas wurde für alle Messreihen stets das gleiche Gasgemisch von Linde verwendet. Die Zusammensetzung ist in Tabelle 6.2 zusammengefasst.

Tabelle 6.2: Prüfgas

	Sollwert	Ist-Wert	relative Messsicherheit
Methan	100 ppm	99,1 ppm	$\pm 2\%$
Stickstoff	Rest		

Die verwendete Messzelle wurde im Auftrag des Heinrich-Blasius Instituts angefertigt und nachträglich für die Montage der Leiterplatte des MEMS-Mikrofons modifiziert, die in Abbildung zu sehen ist. Die Konstruktionszeichnung ist im Anhang angefügt.

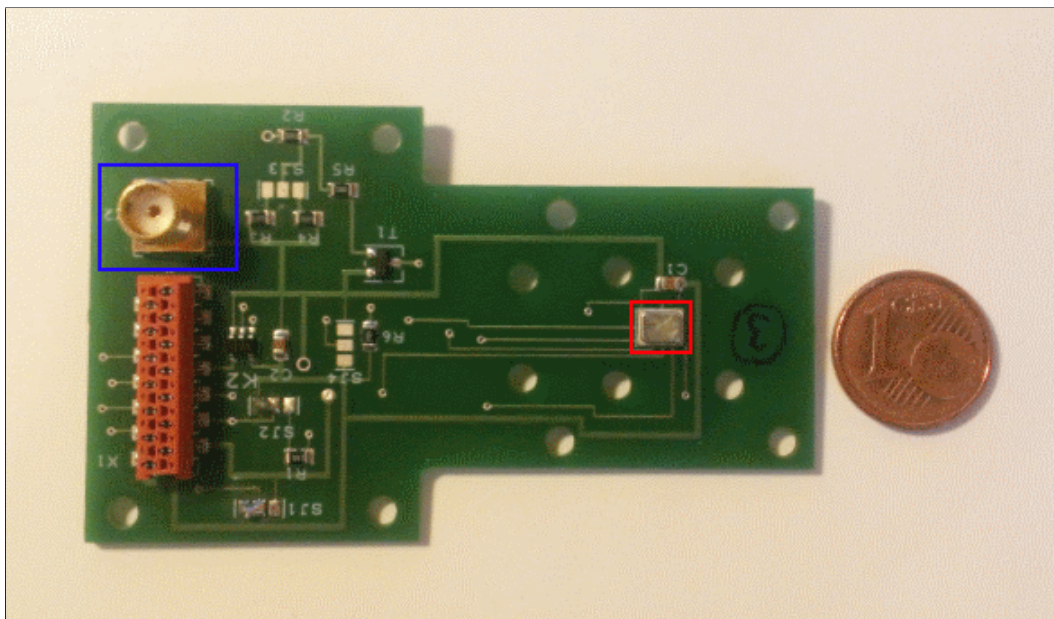


Abbildung 6.1: Im Heinrich-Blasius Institut entstandene Leiterplatte mit dem montierten MEMS-Mikrofon ADMP441 (rot umrandet) und dem Kontakt für das Chopper-Signal (blau umrandet)

## 6.2 Versuchsdurchführung

### 6.2.1 Allgemeines und zur Lichtquelle

Zum Zeitpunkt der Anfertigung dieser Arbeit war die Automatisierung des Messplatzes und im speziellen eine stabile Variation der Wellenlänge des Lasers noch nicht möglich. Daher konnte auch noch kein vollständiges Spektrum des Probengases aufgenommen werden. Messungen bei einer konstanten Wellenlänge waren ebenfalls nicht immer möglich, wie in Abschnitt 6.4 genauer erläutert wird.

Aufgrund dessen wurde als veränderlicher Parameter die optische Ausgangsleistung gewählt. Sie wurde als Funktion der Eingangsleistung des Lasers, soweit es möglich war, in 10 mW Schritten so lange reduziert, bis die Eingangsleistung zu gering war und die Oszillation des Lasers zusammenbrach. In Fällen, wo es durch die veränderte Eingangsleistung zu einem Sprung der Wellenlänge und infolgedessen zu einem Einbruch des Messsignals gekommen ist, wurde über den Etalon-Winkel des optisch-parametrischen Oszillators ( kurz OPO ) manuell ein neuer Peak im Spektrum des Messgases gesucht. Die Eingangsleistung des Lasers wurde immer von einem Startwert ausgehend reduziert.

Wurde ein zu großer Schritt gewählt und ist damit die optische Leistung zu stark gesunken, musste der gewünschte Punkt von einem teilweise deutlich höheren Startwert der Eingangsleistung wieder von oben angefahren werden. Dieses Vorgehen war nötig, da der Laser teilweise auf einen bestimmten Wert eingerastet ist und diesen erst bei einer deutlich höheren Eingangsleistung wieder verlassen hat.

## 6.2.2 Anmerkungen zur Messung mit der USB Mikrofon Firmware

Für qualitative Messungen und für den Vergleich des Elektret- und des MEMS-Mikrofons, wurde die in Abschnitt 5.2.1 beschriebene USB Mikrofon Firmware auf den Mikrocontroller geladen. Die Messdaten des MEMS-Mikrofons wurden über einen Windows PC mit der Software Audacity [25] aufgenommen und als Audiodatei in dem Format WAV gespeichert. Dieses Format speichert Audiodaten unkomprimiert mit 16 Bit. Wie beschrieben wird das photoakustische Signal von dem im MEMS-Mikrofon integrierten Analog-Digital-Wandler mit 24 Bit digitalisiert. Über die USB Schnittstelle werden die Daten ebenfalls mit 24 Bit übertragen. Die Speicherung der Messdaten in 16 Bit verursacht daher eine Verringerung der Dynamik. Ein effektiver Signalverlust ist durch das Abschneiden der letzten 8 Bit nicht zu erwarten, da diese wie in 3.1.3 vom Rauschteppich des MEMS-Mikrofons bedeckt sind.

Damit sowohl die Messsignale des MEMS-Mikrofons, als auch die des Elektret-Mikrofons in einer gemeinsamen Umgebung analysiert und verglichen werden können, wird das Signal des Kondensator-Mikrofons nach dem Mikrofonvorverstärker über einen variablen Spannungsteiler in eine Soundkarte an einem PC eingespeist und digitalisiert. Die dafür verwendete Soundkarte ist die Sound Blaster X-Fi Surround 5.1 Pro USB von Creative. Der Spannungsteiler ist notwendig, um eine Übersteuerung des Eingangs der Soundkarte zu verhindern.

Die Messsignale wurden jeweils mit einer Abtastfrequenz von  $F_s = 48kHz$  aufgenommen. Jede Aufnahme ist 6 Sekunden lang, um für die folgenden Untersuchungen ein möglichst großes Betrachtungsfenster zu haben. Die Aufzeichnungen wurden manuell parallel gestartet und gestoppt, womit ein geringer, zeitlicher Versatz nicht auszuschließen ist. Die tiefste Abtastfrequenzen der Soundkarte beträgt  $F_s = 44,1kHz$ . Da die Goertzel-Algorithmus Firmware mit einer Abtastfrequenz von  $8kHz$  arbeiten soll und dieser Wert mit den  $48kHz$



ein ganzzahliges Teilverhältnis hat, wird diese hohe Abtastfrequenz genutzt und nachträglich in Matlab mit dem Befehl `decimate(Signal,Reduktionsfaktor)` in der Abtastfrequenz auf  $8\text{kHz}$  reduziert. Der Matlab Dokumentation ist zu entnehmen, dass diese Funktion einen Chebyshev Typ I Tiefpass-Filter achter Ordnung einsetzt, um bei der Reduzierung der Abtastrate Aliasing-Effekte zu verhindern.

So können Untersuchungen der Signale unter gleichen Bedingungen gemacht werden und der Einfluss höherfrequenten Rauschens ausgeschlossen werden. Listing 6.1 zeigt den dafür notwendigen Ausschnitt des Quellcodes.

Listing 6.1: Reduzierung der Abtastfrequenz in Matlab

```
1 clear all;

3 % Daten einlesen , Fs = 48 kHz
4 [mems, Fs] = wavread('29-01-2013/01.wav');

6 % Abtastfrequenz reduzieren , Fs = 8 kHz
7 mems_deci = decimate(mems,6);
```

Um beide Signalpfade trotz unterschiedlichen Verstärkungsfaktoren vergleichen zu können, wurde über den Spannungsteiler versucht beide Signale auf einen gemeinsamen Pegel abzugleichen. Diese Einstellung konnte nur sehr grob vorgenommen werden. Für eine Feinabstimmung wurden zu Beginn die arithmetischen Mittelwerte gleichlanger Messsignale gebildet. Das Messsignal besteht aus einem sinusförmigen Anteil, der mit Rauschen belegt ist. Es ist im Idealfall zu erwarten, dass der Mittelwert des sinusförmigen Signals null ist und dass sich bei genügend langen Messsignalen auch das Rauschen im Mittel auflöst. Durch den leakage effect [4, S. 188] können unter anderem hier von allerdings Abweichungen entstehen. Aus diesem Grund wird von den Messsignalen zunächst jeweils der arithmetische Mittelwert  $\hat{x}$  gebildet und dann jeweils von den Messsignalen subtrahiert.

Anschließend wurde der Effektivwert  $x_{eff}$  der jeweiligen Messsignale und dann der Quotient der Effektivwerte des Messsignal des MEMS-Mikrofons und des Elektret-Mikrofons gebildet

$$k = \frac{x_{eff,MEMS}}{x_{eff,Elektret}}. \quad (6.2.1)$$

Da die Verstärkung im Signalpfad des Elektret-Mikrofons größer war, gilt  $k < 1$ . Das Messsignal des Elektret-Mikrofons wurde dann mit  $k$  multipliziert, um so eine Annäherung der Signalamplituden beider Messpfaden zu erreichen. Diese Rechnungen wurden mit einem Messsignal durchgeführt, bei dem der SNR am größten ist um so den Einfluss des Rauschens möglichst gering zu halten. Dabei hat sich ein Wert von  $k = 0,49$  ergeben. Die Bestimmung von  $k$  wurde für andere Messsignale mit sinkender Signalstärke ebenfalls durchgeführt. Als arithmetischer Mittelwert über jeweils 10 Messsignale hat sich der Wert  $k = 0,53$  ergeben, der im Folgenden auch verwendet werden soll.

### 6.2.3 Anmerkungen zur Messung mit der Goertzel-Algorithmus Firmware

Bei den Messungen mit der Goertzel-Algorithmus Firmware wurden zwei Schritte durchgeführt. Im ersten Schritt wurden jeweils 43 Ausgangswerte des Goertzel-Algorithmus bei unterschiedlichen Längen  $N$  der Eingangssequenz und sonst festgehaltenen Parametern des Messaufbaus aufgenommen. Dieser Wert stellt ein gutes Stichprobenintervall dar. Dadurch sollte ein optimales  $N$  gefunden werden. Optimal bedeutet hier, dass ein stabiles Ausgangssignal bei einer möglichst geringen Bandbreite erzeugt wird.

Im zweiten Schritt wurde mit der gewählten Länge  $N$  der Eingangssequenz eine Messreihe aufgenommen, bei der parallel zur Ausführung des Goertzel-Algorithmus das Messsignal des Kondensator-Mikrofons aufgenommen wurde. Auf diese Daten soll dann der Goertzel-Algorithmus unter Matlab angewendet und das Ergebnis mit dem der Mikrocontroller Firmware verglichen werden.

Tabelle 6.3 zeigt den Mittelwert und die Standardabweichung der Ergebnisse des Goertzel-Algorithmus in Abhängigkeit der Sequenzlänge  $N$  sowie die Frequenzauflösung  $\Delta f$  bei einer Abtastfrequenz von  $F_s = 8kHz$ . Die Messungen wurden dabei auf das jeweilige  $N$  normiert.

Tabelle 6.3: Mittelwert und Standardabweichung der Goertzel-Algorithmus Firmware in Abhängigkeit der Sequenzlänge  $N$

$N$	$\Delta f$	Mittelwert	Standardabweichung	Abweichung vom Mittelwert in %
2000	4 Hz	0,1795	0,0133	7,4
2666	3 Hz	0,189	0,0164	8,7
4000	2 Hz	0,1574	0,0266	16,9
6000	1,33 Hz	0,0877	0,0398	45,4
8000	1 Hz	0,0607	0,0332	54,6

Die Ergebnisse in Tabelle 6.3 zeigen, dass die Streuung um den Mittelwert um so größer wird, je besser die Frequenzselektion des Goertzel-Filters wird. Sehr anschaulich verdeutlicht dies Abbildung 6.2. Bis auf einen stärkeren Ausreißer an neunter Stelle, bewegen sich die Ausgangswerte für  $N = 2000$  innerhalb eines geringen Streubereichs, wie auch der Tabelle 6.3 zu entnehmen ist.

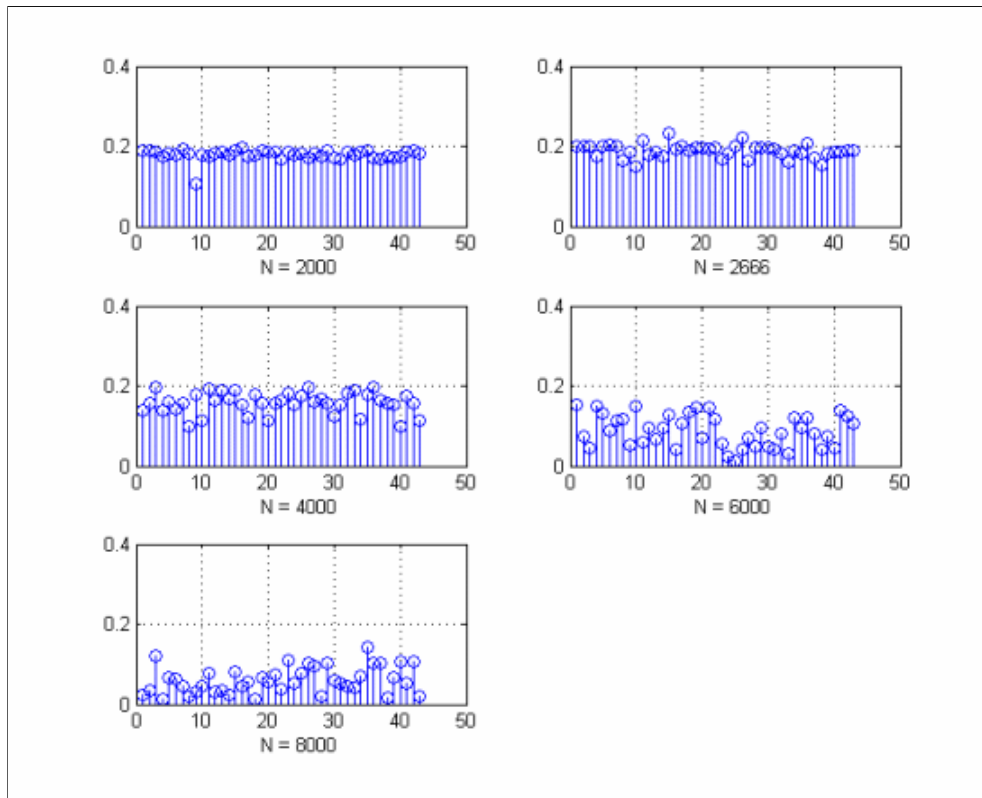


Abbildung 6.2: Ergebnisse des Goertzel-Algorithmus in Abhängigkeit der Sequenzlänge  $N$

Für größer werdende  $N$  wird der Goertzel-Filter zwar immer schmalbandiger, die selektierte Frequenzkomponente scheint sich aber immer wieder aus dem Durchlassbereich des Filters herauszubewegen. Dadurch entstehen größere Schwankungen am Filterausgang und es verringert sich der Beitrag am Ausgangssignal, wie der Vergleich der Mittelwerte in Tabelle 6.3 für  $N = 6000$  und  $N = 8000$  gegenüber  $N = 2000$  zeigt.

Um stabile Ausgangswerte des Goertzel-Algorithmus ohne größere Streuungen zu erhalten, ist es sinnvoll, mit einer geringeren Bandbreite zu Arbeiten. Dies hat auch den Vorteil, dass die Messdauer kleiner wird. Wenn schmalbandiger gemessen werden soll, kann die Streuung der Werte bis zu einem gewissen Grad durch einen nachgeschalteten digitalen Tiefpass in Form einer Mittelwertbildung abgefangen werden.

## 6.3 Experimentelle Ergebnisse

### 6.3.1 Messungen mit der USB Mikrofon Firmware

Die weitere Analyse der Messsignale aus beiden Signalpfaden wird im Frequenzbereich vorgenommen. Speziell wird auch nur der Bereich der positiven Frequenzachse betrachtet. Der Informationsgehalt der negativen Frequenzachse ist damit gleichwertig.

Da es durch die begrenzte Auflösung des DFT Spektrums und den nicht vermeidbaren leakage effect zu einer Verschmierung des harmonischen Signals im Messsignal kommt, ist es wenig sinnvoll absolute Werte zu vergleichen. Die gemachten Messungen und die daraus entstandenen Bilder sind nur Momentaufnahmen. Durch die in 6.2.2 gemachten Anpassungen beider Signalpfade kommt es aber zu sehr ähnlichen Spektralampplituden, wie man beispielsweise in Abbildung 6.3 sehen kann.

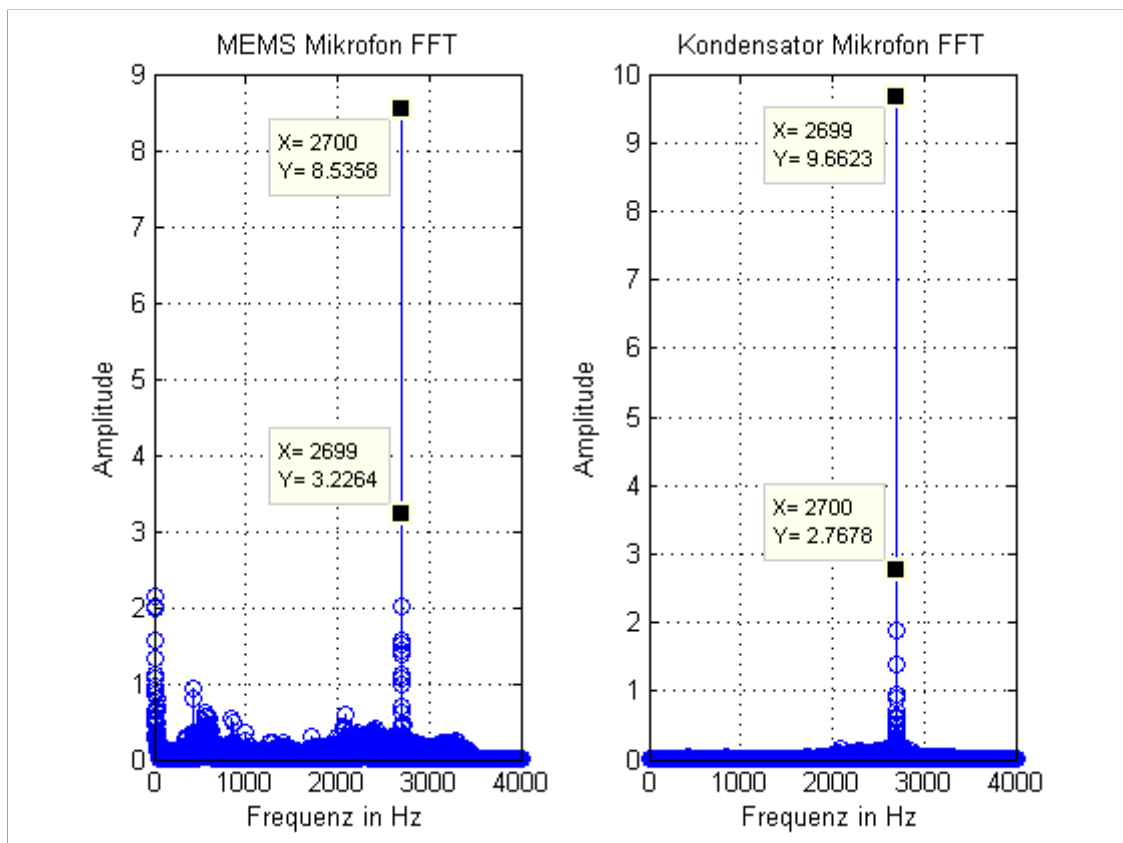


Abbildung 6.3: Gegenüberstellung der Amplitudenspektren beider Signalpfade bei 134,90mW optischer Ausgangsleistung und einer Wellenlänge von 3315,837nm. Abtastfrequenz  $F_s = 8kHz$ , Länge der FFT  $N = 8000$ , nicht normiert.

Bei näherungsweise gleichen Verstärkungsfaktoren in beiden Signalpfaden sind daher mit beiden Mikrofonen ähnliche Signalstärken messbar. Das schlechtere Signal-Rausch-Verhältnis des ADMP441,  $61\text{dBA}$ , gegenüber dem des Primo EM158,  $74\text{dBA}$ , macht an dieser Stelle noch keinen Unterschied der beiden Mikrofone aus. Relevanter wird der deutliche Unterschied von  $11\text{dB}$  erst bei sehr geringen Signalstärken, wie sie in Abbildung 6.4 zu sehen sind. Hier ist zu erkennen, dass der Rauschteppich des MEMS-Mikrofons, über die betrachtete Bandbreite, deutlich stärker ausgeprägt ist, als der des Elektret-Mikrofons. Ungünstig ist bei dem Elektret-Mikrofon, dass sich das Rauschen um den Anteil des photoakustischen Signals im Frequenzspektrum lokalisiert.

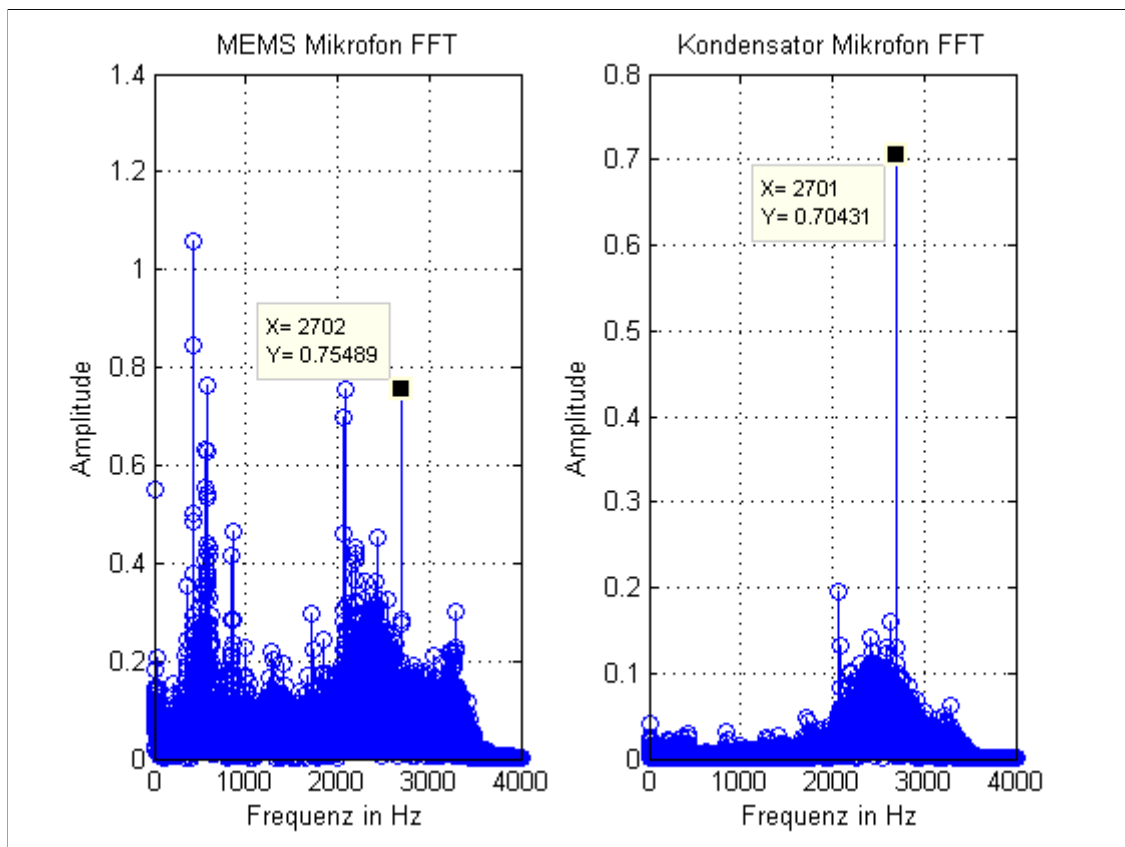


Abbildung 6.4: Gegenüberstellung der Amplitudenspektren beider Signalpfade ohne Laseranregung. Abtastfrequenz  $F_s = 8\text{kHz}$ , Länge der FFT  $N = 8000$ , nicht normiert.

Zoomt man in den relevanten Frequenzbereich hinein, wie es in Abbildung 6.5 zu sehen ist, dann sind die momentanen Rauschamplituden um die eigentliche Nutzfrequenz herum im Mittel gering genug, so dass noch ein deutlicher Peak zu erkennen ist. Dies ist aber nur eine Momentaufnahme.

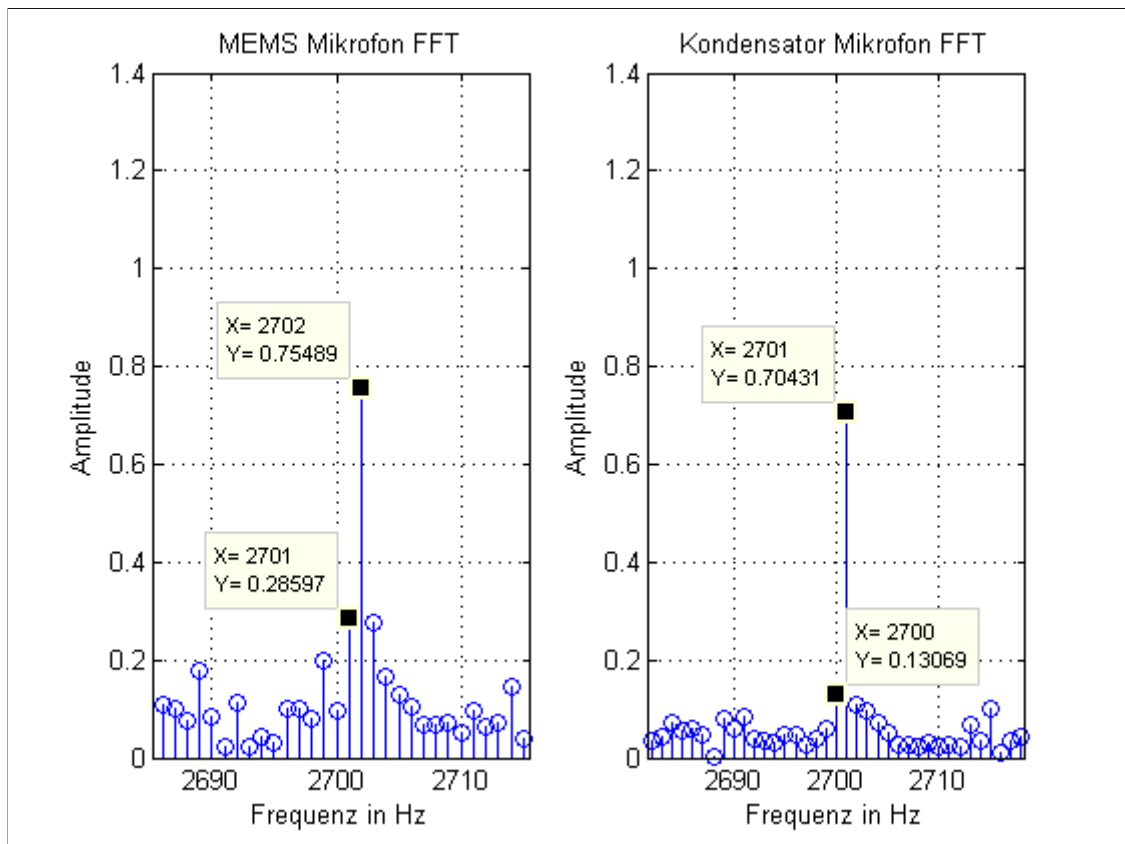


Abbildung 6.5: Gegenüberstellung der Amplitudenspektren beider Signalpfade bei  $134,90\text{mW}$  optischer Ausgangsleistung und einer Wellenlänge von  $3315,837\text{nm}$ . Abtastfrequenz  $F_s = 8\text{kHz}$ , Länge der FFT  $N = 8000$ , nicht normiert.

Es muss bei allen Untersuchungen berücksichtigt werden, dass trotz abgeschaltetem Laser immer noch eine deutliche Anregung der Spektralkomponente der Chopper-Frequenz zu messen ist. Dies ist auf die nicht ideale akustische Abschirmung der Messzelle zurück zu führen. Als Scheinsignal leistet also bei allen Messungen das Geräusch des Chopper-Rades einen Beitrag zum Messsignal. Dieser bewegt sich bei beiden Mikrofonen ebenfalls innerhalb der selben Größenordnung, wie man in Abbildung 6.4 und 6.5 erkennen kann.

Um konkretere Leistungsprognosen für beide Mikrofone anzustellen, empfiehlt es sich daher Maßnahmen zur akustischen Abschirmung der Messzelle zu ergreifen. Dann kann genauer untersucht werden, wie stark die Rauschcharakteristik im Signal des MEMS-Mikrofons tatsächlich systematisch ist und ob noch geringere Signalstärken detektiert werden können.

### 6.3.2 Messungen mit der Goertzel-Algorithmus Firmware

Ausgehend von den Vorüberlegungen in Abschnitt 6.2.3, wurde  $N = 2000$  gewählt. Die Messreihe wurde entsprechend dem Vorgehen in Abschnitt 6.2.1 aufgenommen. Von einem gewählten Maximum wurde schrittweise die Eingangsleistung des Lasers reduziert, um die gewählte Abstufung der optischen Ausgangsleistung zu erreichen. Die Aufnahmen mit dem Elektret-Mikrofon waren wieder mindestens 6 Sekunden lang und wurden ebenfalls nachträglich in Matlab in der Abtastfrequenz von  $F_s = 48kHz$  auf  $F_s = 8kHz$  reduziert. Diese wurden durch ein Matlab Skript, das den Goertzel-Algorithmus in derselben Weise wie die Goertzel-Algorithmus Firmware implementiert, verarbeitet. Die Messsignale des Elektret-Mikrofons wurden wie in Abschnitt 6.2.2 ebenfalls mit dem Faktor  $k = 0,53$  gewichtet, um die unterschiedliche Verstärkung in beiden Signalpfaden größtenteils auszugleichen.

In diesem Abschnitt geht es prinzipiell um den Vergleich der Messergebnisse des Goertzel-Algorithmus Firmware und dem des Lock-In Verstärker. Die Aufgabenstellung fragt diesbezüglich, ob eine Erhöhung der Nachweisempfindlichkeit des photoakustischen Signals durch das neue Mikrofon und den neuen Ansatz der Signalverarbeitung erreicht wird. Berücksichtigt man die Erkenntnis aus dem vorherigen Abschnitt, dass das Geräusch des Chopper-Rades einen messbaren Anteil an dem photoakustischen Signal beiträgt, so ist kein absolutes Minimum der Signalstärke auszumachen beziehungsweise der als Minimum angenommene Punkt ist kein photoakustisches Signal mehr. Betrachtet man die aufgenommenen Messsignale im Frequenzspektrum, dann waren die Spektralwerte bei der Chopper-Frequenz nie so klein, dass sie im Rauschen untergegangen sind, wie in Abbildung 6.4 und speziell in Abbildung 6.5 zu sehen war.

Die Tabelle 6.4 auf der nachfolgenden Seite fasst die Mittelwerte  $\bar{x}$  der Ergebnisse der Goertzel-Algorithmus Firmware, des Goertzel-Algorithmus in Matlab angewendet auf die Aufnahmen des Kondensator-Mikrofons und die Ausgangsspannung des Lock-In Verstärkers zusammen. Der arithmetische Mittelwert wurde dabei über 29 Messwerte bestimmt. Weiter werden die momentanen Wellenlängen und die optische Ausgangsleistungen angegeben.

Die Ausgangsspannung des Lock-In Verstärkers unterliegt der Verstärkung der integrierten Verstärkerstufen, so dass die in Tabelle 6.4 gezeigten Ausgangswerte nicht als Absolutwerte miteinander verglichen werden können. Es soll viel mehr eine grundsätzliche Tendenz aufgezeigt werden, die das Verfahren des Lock-In Verstärkers und das des Goertzel-Algorithmus gegenüber stellt.

Tabelle 6.4: Mittelwerte der Goertzel-Algorithmus Firmware und des Goertzel-Algorithmus in Matlab für  $N = 2000$  und des Lock-In Verstärkers sowie die gemessene Wellenlänge, optische Leistung und Chopper-Frequenz. Zelldruck: 1015 mbar

Aufnahme	$\bar{x}$ Firmware	$\bar{x}$ Matlab	$\bar{x}$ Lock-In (mV)	Wellenlänge (nm)	optische (mW)	Chopper-Frequenz (Hz)
1	3,67	3,48	36,5	3315,146	136,0	2699
2	3,25	3,24	31,2	3315,151	125,1	2701
3	2,87	2,70	27,9	3315,153	116,8	2698
4	2,97	2,56	28,3	3315,066	106,6	2698
5	2,43	2,36	24,1	3315,067	94,3	2699
6	1,44	1,27	12,7	3314,957	81,1	2697
7	1,00	0,91	8,8	3315,222	72,9	2699
8	1,12	1,05	10,2	3315,181	63,3	2700
9	0,54	0,52	4,1	3315,335	53,0	2700
10	0,30	0,24	2,5	3315,336	46,7	2700
11	0,26	0,09	0,8	Laser aus	Laser aus	2700
12	0,18	0,11	0,8	Laser aus	Laser aus	2700



Die zweite beziehungsweise dritte und vierte Spalte in Tabelle 6.4 sollte an dieser Stelle nicht falsch interpretiert werden. Die Zahlen mögen zwar vermuten lassen, dass es sich hier um gleiche Werte handelt, die sich lediglich um den Faktor 10 unterscheiden. Es ist jedoch weder bekannt, welche Spannungsverstärkung das MEMS-Mikrofon vor dem integrierten AD-Wandler besitzt, noch welche Gesamtspannungsverstärkung in der Messkette des Elektret-Mikrofons wirkt. Näheres hierzu muss in den weiterführenden Untersuchungen des Messaufbaues geklärt werden.

Die Abbildungen 6.6 und 6.7 veranschaulicht noch einmal anhand von realen Messdaten den Frequenzgang des Goertzel-Filters, wie er in Abbildung 4.5 skizziert wurde.

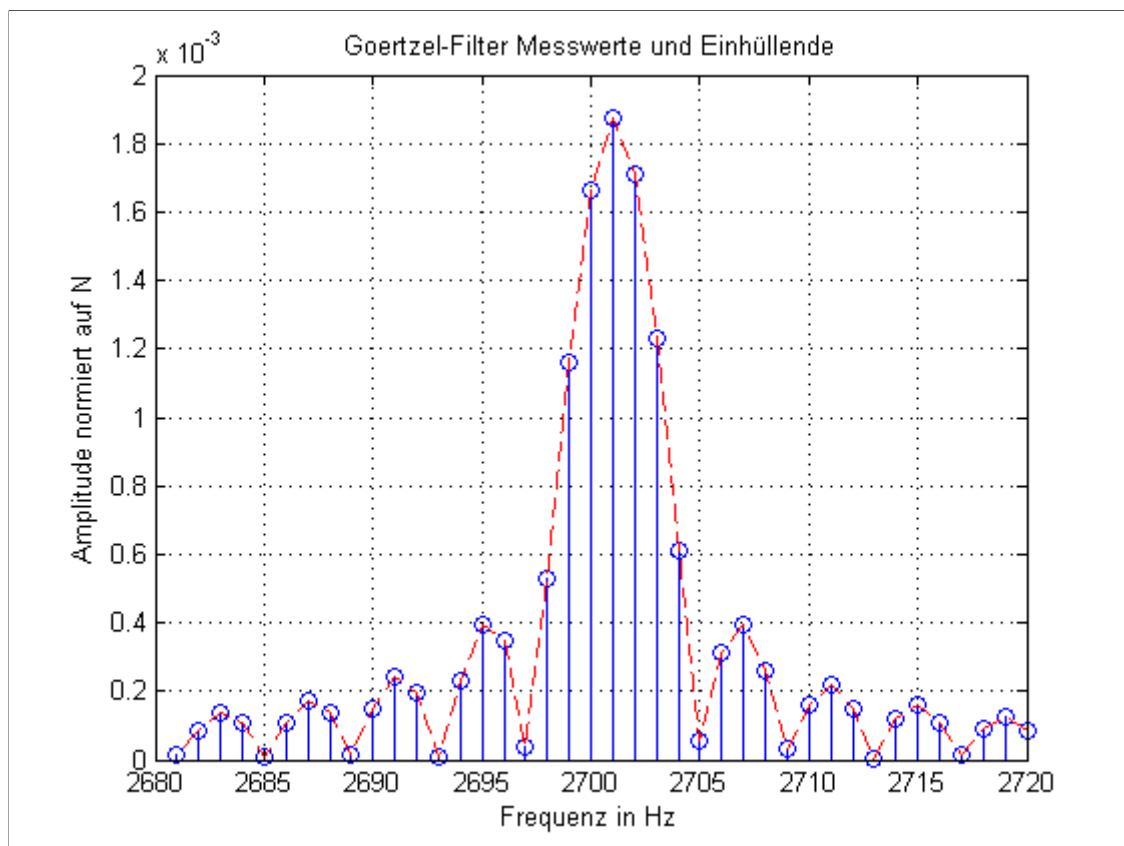


Abbildung 6.6: Frequenzgang des Goertzel-Filters (rot) mit realen Messdaten (blau),  $N = 2000$ , normiert

Es ist in Abbildung 6.6 sehr gut der Verlauf des Dirichlet-Kerns zu erkennen, welcher die Fouriertransformierte des diskreten Rechteck-Fensters darstellt, mit dem das Messsignal durch die endliche Messdauer gewichtet wurde. Weiter lässt sich im Vergleich mit Abbildung 6.7 erkennen, dass der Goertzel-Filter, mit steigender Zahl der berücksichtigten Ab-

tastwerte, immer schmalbandiger im Durchlassbereich wird. Beide Abbildungen wurden mit Messsignalen und der Implementierung des Goertzel-Algorithmus in Matlab erzeugt.

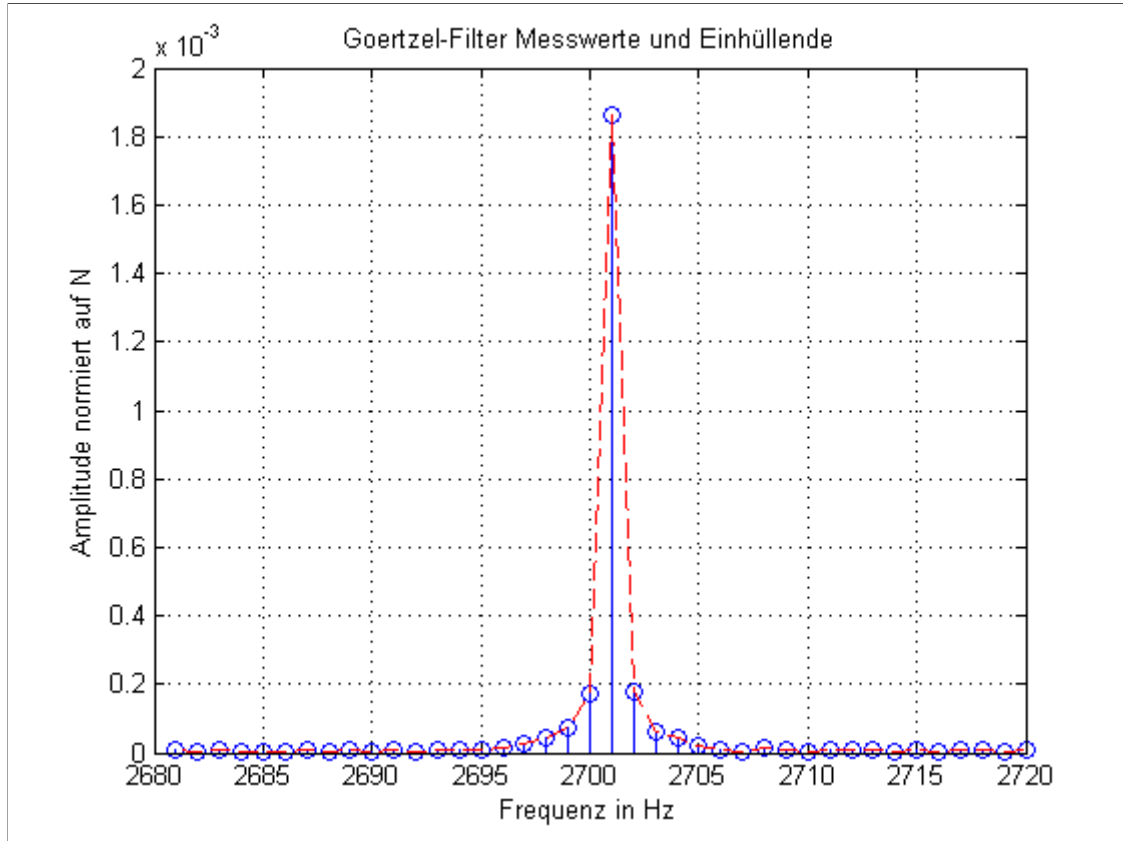


Abbildung 6.7: Frequenzgang des Goertzel-Filters (rot) mit realen Messdaten (blau),  $N = 8000$ , normiert

## 6.4 Messtechnische Probleme

Bei der Aufnahme der gemachten Messreihen wurde deutlich, wie empfindlich das Verfahren der photoakustischen Spektroskopie gegenüber Abweichungen im Messaufbau ist. Nachfolgend sollen einige Punkte besprochen werden, die, sowohl im Speziellen für den gegebenen Messaufbau als auch im Allgemeinen, zu berücksichtigen sind und die Messungen teilweise stark beeinflusst haben.

In Abschnitt 2.1 wurde erwähnt, dass das photoakustische Signal eine Verstärkung erfahren kann, wenn die Chopperfrequenz auf die Resonanzfrequenz der Messzelle abgestimmt wird. Die Resonanzfrequenz der Zelle kann sich aufgrund von Veränderung der Umgebungstemperatur und der Erwärmung durch Lichtstreuung innerhalb der Zelle verschieben. Dies muss

eventuell durch die Anpassung der Chopperfrequenz berücksichtigt werden, da der Einfluss der Resonanzverschiebung schon bei wenigen Hertz an der Signalstärke zu erkennen ist.

Im Folgenden ist das Chopper-Rad selber zu nennen. Auf Grund intensiver Beobachtungen konnte festgestellt werden, dass die Drehfrequenz teilweise deutlich von der eingestellten Frequenz weg driftet. Abbildung 6.8 verdeutlicht dieses Verhalten. Das Bild zeigt das Frequenzspektrum einer sechs Sekunden langen Aufzeichnung. Anstelle einer scharfen Linie erkennt man deutlich die Wanderung der Chopper-Frequenz.

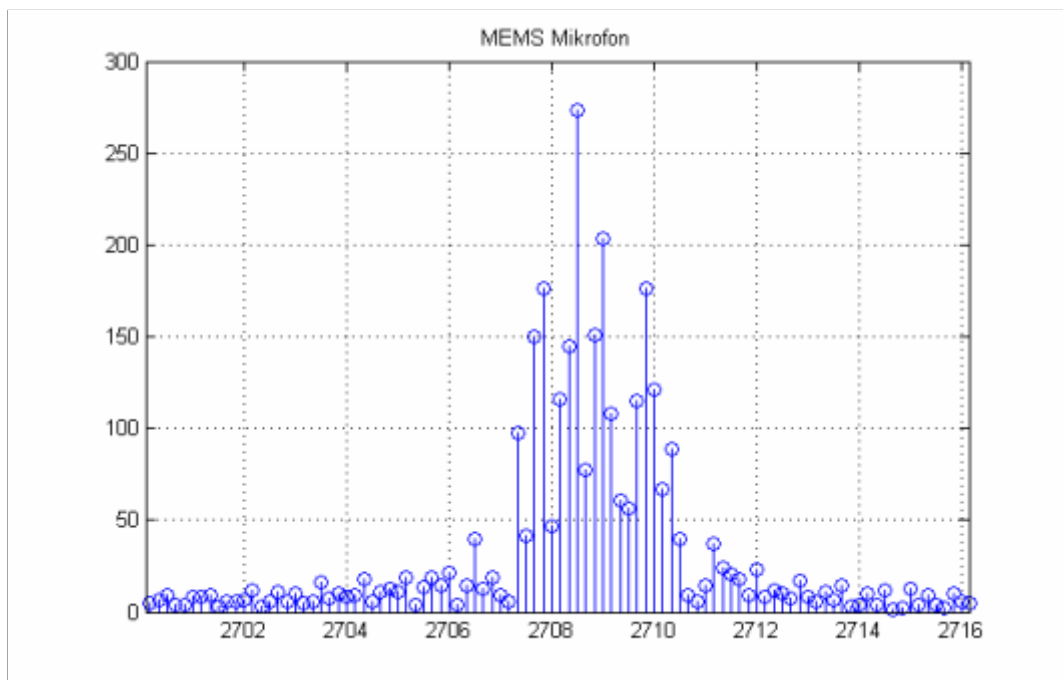


Abbildung 6.8: Schwankung der Chopper-Frequenz

An der Linienlänge kann man erkennen, wie lange die jeweilige Frequenz eingenommen wurde. Eingestellt war eine Chopper-Frequenz von etwa  $2708\text{kHz}$ . Die größte Schwankung kann man dem Bild dabei mit über 2 Hz zu entnehmen.

Des Weiteren ist die Stabilität der Wellenlänge des verwendeten Lasers zu berücksichtigen. Durch die Veränderung der Eingangsleistung des Lasers ist es immer wieder zu Sprüngen der Wellenlänge gekommen. Eine Folge dessen war, dass das photoakustische Signal teilweise massiv in der Signalstärke eingebrochen ist, weil die Laserwellenlänge nicht mehr der Absorptionslinie des Probengases entsprochen hat. Teilweise konnte es auch beobachtet werden, dass Sprünge von einigen hundert Pikometern in der Wellenlänge ohne jede Änderung der Lasereingangsleistung aufgetreten sind. Dies sollte speziell bei der Aufnahme von ganzen photoakustischen Spektren berücksichtigt werden.

## 6.5 Diskussion der Implementierungsarten des Goertzel-Algorithmus

Die Goertzel-Algorithmus Firmware wurde unter der Verwendung von Fließkomma Datentypen mit einfacher Genauigkeit (Datentyp *float* in C) implementiert. Dies hat auf dem ARM Cortex M3, wie in Abschnitt 3.2 beschrieben, den Nachteil, dass alle Operationen vom Compiler in Software emuliert werden müssen, da keine Floating Point Unit hardwareseitig zur Verfügung steht. Der eigentliche Ansatz war eine Implementierung des Algorithmus mit Festkomma Datentypen.

Da der Goertzel-Algorithmus als Filter seine Polstellen auf dem Einheitskreis besitzt, kann der Filter instabil werden. Dies kann schnell passieren, wenn die Filterkoeffizienten zu grob quantisiert, also mit zu wenig Bits dargestellt werden. Neben der Instabilität des Filters kann es dadurch auch zu einem ganz anderen Filterverhalten kommen, als es bei der Dimensionierung vorgesehen war. Dann ist der Filter zwar stabil, aber der Frequenzgang des Filters kann vollkommen anders sein. Eine Analyse zu dieser Problematik im Allgemeinen für IIR Filter 2. Ordnung wird, unter anderem, in [20, S. 929f] und im Speziellen für den Goertzel-Filter in [26] vorgenommen.

Ein weiteres Problem sind Überläufe und Unterläufe, nachfolgend einfach Überläufe, in Folge der Summation in Gleichung 4.3.16. Mit Zahlen in Festkomma-Darstellung ist es schwer einen Kompromiss aus der Genauigkeit der Zahlendarstellung und der Vermeidung von Überläufen zu finden. Versucht wurde der Ansatz einer 8.24 Darstellung, die in 32 Bit *long int* gespeichert wird. Diese Darstellung besitzt 1 Vorzeichenbit, 7 Vorkommabits und 24 Nachkommabits. Die Abtastwerte des MEMS-Mikrofons sollten demnach als die 24 Nachkommabits interpretiert werden. Die 7 Vorkommabits, mit denen sich die Zahlen  $-2^7 = -128$  bis  $(2^7 - 1) = 127$  darstellen lassen, hätten ausreichen sollen, um die erwarteten kleinen Zahlenwerte aufgrund der geringen Signalamplituden aufsummieren zu können; ohne die Gefahr von Überläufen.

Da der Kosinus auf den Bereich  $-1 \leq x \leq 1$  beschränkt ist, konnte der Koeffizient  $a_1/2$  des IIR Teils des Goertzel-Filters im 1.31 Format dargestellt werden. Um ein zusätzliches Bit für die Darstellung des Kosinuswertes an der Stelle  $\cos(\pi f_k/F_s)$  zu gewinnen, wurde die Multiplikation mit Zwei erst an entsprechender Stelle vorgenommen.

Multiplikationen von Zahlen im 8.24 und 1.31 Format führen zu einem 64 Bit (*long long int* in C) breiten Ergebnis im 9.55 Format, wobei die obersten beiden Bits das Vorzeichen beinhalten. Durch entsprechenden Shift-Operationen wurde das Ergebnis wieder in das 8.24 Format gebracht, was einen Verlust der Darstellungsgenauigkeit zur Folge hatte.

Der so implementierte Goertzel-Filter wurde dann unter denselben Bedingungen wie in Abschnitt 6.2.3 erprobt. Ergebnis dessen waren nicht nachzuvollziehende Schwankungen des

Ausgangssignals. 40 Ausgangswerte des Goertzel-Filters, die auf  $2^{31} - 1$  normiert wurden, sind in Abbildung 6.9 zu sehen.

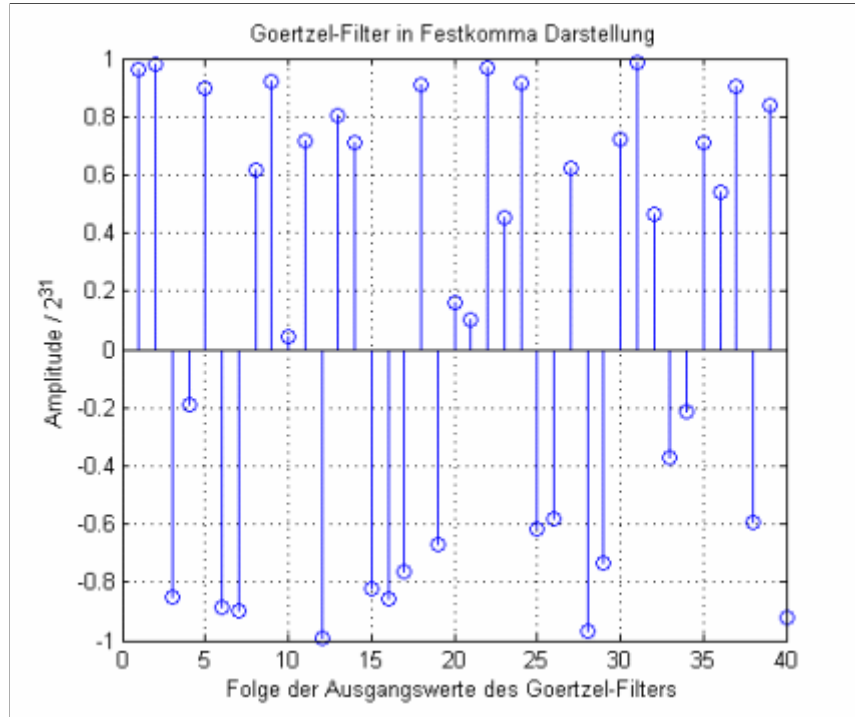


Abbildung 6.9: Unkontrollierte Schwankungen am Ausgang des Goertzel-Filters in einer Festkomma-Implementierung

Die Sprünge großer positiver Zahlen hin zu großen negativen Zahlen und umgekehrt lassen vermuten, dass es zu Überläufen innerhalb der Summationen der Abtastwerte gekommen ist. Das Problem konnte beispielsweise mit der Skalierung der Abtastwerte nicht behoben werden. Auch Anpassungen gemäß [26] haben das Verhalten der Software nicht geändert.

Probehalber wurde der Goertzel-Filter mit Messdaten, die in 6.3.1 gewonnen wurden, auf dem STM32f4-Discovery Evaluation Board von ST Microelectronics implementiert. Dieses Evaluation Board basiert auf einem ARM Cortex M4 mit Hardware FPU und 1 Megabyte Flash-Speicher. In diesen konnten die Messdaten geladen werden und mit dem Goertzel-Algorithmus verarbeitet werden. Die Ergebnisse für gegebene Messdaten waren bis auf die vierte Nachkommastelle identisch mit den Berechnungen in Matlab. Dies war der Grund, weswegen die Implementierung auf dem Stellaris LM3S9B92 doch mit Fließkomma Arithmetik realisiert wurde. Eine Umstellung der gesamten Goertzel-Algorithmus Firmware auf das STM32f4-Discovery war zu diesem späten Zeitpunkt nicht mehr vollständig möglich.

## 7 Zusammenfassung und Ausblick

Die Aufgabenstellung dieser Arbeit sah die Inbetriebnahme eines digitalen MEMS-Mikrofons, ADMP441 von Analo Devices, mit einem geeigneten Mikrocontroller sowie die Entwicklung einer Software zur digitalen Signalverarbeitung der Messdaten innerhalb eines photoakustischen Spektrometers vor. Der in der Arbeit verwendete Stellaris LM3S9B92 Mikrocontroller, der auf dem ARM Cortex M3 basiert, hat sich wegen seiner Leistungsfähigkeit und den zur Verfügung stehenden Schnittstellen als für den Verwendungszweck geeignet herausgestellt.

Für den verwendeten Mikrocontroller sind zwei Softwareprojekte entstanden. Zum einen wurde auf der Basis eines bereits existierenden Beispielprojekts von Texas Instruments eine Software entwickelt, die die Übertragung der mit dem ADMP441 aufgenommenen Daten an einen PC über die USB 2.0 Schnittstelle des Stellaris LM3S9B92 ermöglicht. Diese Software war grundlegend für alle weiteren Untersuchungen der Messsignale innerhalb dieser Arbeit und hatte damit einen hohen Stellenwert bei der Bearbeitung dieser Aufgabenstellung. Zum anderen ist ein Softwareprojekt entstanden, das den Goertzel-Algorithmus zur gezielten Detektion eines photoakustischen Signals innerhalb eines mit Rauschen belegten Messsignals nutzt.

Es ist ganz klar festzuhalten, dass das in dieser Arbeit verwendete Elektret-Mikrofon technisch sehr ausgereift und dass der verwendete Lock-In Verstärker deutlich Leistungsstärker ist, als der Stellaris LM3S9B92 Mikrocontroller. In dem Lock-In Verstärker arbeiten zwei digitale Signalprozessoren, ein dedizierter digitaler Signalgenerator und ein Mikrocontroller für die allgemeine Steuerung [27]. Dem gegenüber steht ein einzelner Mikrocontroller, der die komplette Datenverarbeitung und die Steuerung der Geräteschnittstellen gleichzeitig übernehmen muss. Berücksichtigt man damit die Einfachheit des entwickelten Systems, so ist dessen Ansatz und Weiterentwicklung unter dem Hintergrund der Kostenreduzierung, Miniaturisierung und der damit verbundenen Massentauglichkeit besonders interessant.

In dieser Arbeit wurde nur ein möglicher Lösungsweg vorgestellt. Da die Signalverarbeitung in Software realisiert wurde, kann man sich auch einen grundsätzlich anderen Lösungsansatz vorstellen. Das System ist dahingehend sehr flexibel. Die Untersuchungen haben aber gezeigt, dass mit dem ADMP441 in Verbindung mit dem Goertzel-Algorithmus grundsätzlich photoakustische Spektroskopie betrieben werden kann. Bei einer stabileren Chopperfrequenz ist eine noch schmalbandigere Signalverarbeitung möglich. Zu diesem Zeitpunkt

kann aber nicht abschließend entschieden werden, ob das Potential dieses neuen und verkleinerten Systems bereits vollständig erschöpft ist oder durch weitere Optimierungen am gesamten Aufbau weitere Leistungssteigerungen möglich sind. Dies haben weiterführende Untersuchungen zu zeigen.

## 7.1 Nachweisempfindlichkeit

Die Auswertungen der Messungen haben gezeigt, dass der begrenzende Faktor bei der Nachweisgrenze des neuen und auch des alten Systems die akustische Abschirmung der Messzelle ist. Ein absolutes Minimum der Signalstärke konnte wegen des Scheinsignals, dass das Chopper-Rad verursacht hat, nicht ermittelt werden. Allerdings war es wohl mit dem Elektret- als auch mit dem neuen MEMS-Mikrofon möglich die geringste Signalstärke zu detektieren.

## 7.2 Weiterführende Untersuchungen

Die Stabilität der Chopper-Frequenz ist ein großes Problem bei schmalbandigen Messungen. Hier wäre es erstrebenswert das Chopper-Rad durch eine stabilere Lösung oder direkt durch einen pulsabaren Laser zu ersetzen. Daneben ist die Optimierung der akustischen Abschirmung der Zelle aus den in dieser Arbeit genannten Gründen dringend zu empfehlen.

Weiter birgt die USB Mikrofon Firmware großes Potential für die Entwicklung von umfangreicheren Analyseapplikationen auf einem PC, da dort der Speicherbedarf und die Rechenleistung um ein vielfaches größer ist, als auf einem Mikrocontroller. Dort kann dann eine individuelle Signalaufbereitung und Signalanalyse vorgenommen.

# Literaturverzeichnis

- [1] MÖSER, Michael: *Technische Akustik*. 7. Berlin and Heidelberg and New York : Springer, 2007 (VDI). – ISBN 9783540713869
- [2] PRIMO: *Electret Condenser Microphone Ultra High sensitivity EM158 Specification*. <http://www.primocorp.co.jp/product/PDF/EM158.pdf> (08.02.2013),
- [3] TEXAS INSTRUMENTS: *Stellaris®LM3S9B92 Microcontroller Data Sheet (2012)*
- [4] MEYER, Martin: *Signalverarbeitung: Analoge und digitale Signale, Systeme und Filter*. 5. Wiesbaden : Vieweg + Teubner, 2009. – ISBN 978-3-8348-0494-5
- [5] ROSENCWAIG, Allan: *Photoacoustics and photoacoustic spectroscopy*. New York : Wiley, 1980. – ISBN 0471044954
- [6] DEMTRÖDER, Wolfgang: *Laserspektroskopie: Grundlagen und Techniken*. 5. Berlin [u.a.] : Springer, 2007. – ISBN 978-3-540-33792-8
- [7] TELLE, Helmut H. ; DONOVAN, Robert J. ; GONZÁLEZ UREÑA, Ángel: *Laser chemistry: Spectroscopy, dynamics and applications*. Chichester and West Sussex and England and Hoboken and NJ : John Wiley & Sons, 2007. – ISBN 978-0-471-48570-4
- [8] GÖRNE, Thomas: *Tontechnik: Schwingungen und Wellen, Hören, Schallwandler, Impulsantwort, Faltung, Sigma-Delta-Wandler, Stereo, Surround, WFS, Regiegeräte, ton-technische Praxis*. 3. München : Hanser, Carl, 2010. – ISBN 3446423958
- [9] BALTES, Henry ; BRAND, Oliver ; FEDDER, Gary K. ; HIEROLD, Christofer ; KORVINK, Jan G. ; TABATA, Osamu: *CMOS-MEMS: Advanced Micro and Nanosystems*. Weinheim : Wiley-VCH, 2005. – ISBN 3527310800
- [10] ANALOG DEVICES: *AN-1003: Recommendations for Mounting and Connecting the Analog Devices, Inc., Bottom-Ported MEMS Microphones*. [http://www.analog.com/static/imported-files/application\\_notes/AN-1003.pdf](http://www.analog.com/static/imported-files/application_notes/AN-1003.pdf) (08.02.2013),
- [11] ANALOG DEVICES: *AN-1112: Microphone Specifications Explained*. [http://www.analog.com/static/imported-files/application\\_notes/AN-1112.PDF](http://www.analog.com/static/imported-files/application_notes/AN-1112.PDF) (08.02.2013),



- [12] LEWIS, Jerad: *Understanding Microphone Sensitivity*. [http://www.analog.com/library/analogdialogue/archives/46-05/understanding\\_microphone\\_sensitivity.pdf](http://www.analog.com/library/analogdialogue/archives/46-05/understanding_microphone_sensitivity.pdf) (08.02.2013),
- [13] NAUNDORF, Uwe: *Digitale Elektronik: Theoretische Grundlagen und Schaltungsanalysen*. München : Oldenbourg, 2004. – ISBN 3486275984
- [14] ANALOG DEVICES: *ADMP441: Omnidirectional Microphone with Bottom Port and I2S Digital Output Data Sheet (10/2012), PDF. Rev. B*. <http://www.analog.com/en/mems-sensors/mems-microphones/admp441/products/product.html> (08.02.2013),
- [15] TEXAS INSTRUMENTS: *Dual-SPI Emulating I2S on Stellaris®LM4F MCUs*. <http://www.ti.com/lit/an/spma042a/spma042a.pdf> (08.02.2013),
- [16] WÜST, Klaus: *Mikroprozessortechnik: Grundlagen, Architekturen, Schaltungstechnik und Betrieb von Mikroprozessoren und Mikrocontrollern ; mit 44 Tabellen*. 4. Wiesbaden : Vieweg+Teubner Verlag / Springer Fachmedien Wiesbaden GmbH Wiesbaden, 2011. – ISBN 3834809063
- [17] KEIL, Reinhard: *Digital Signal Processing with Cortex-M Microcontrollers*. [http://www.icmagazineonline.com/current/pdf/Pg28-29-IQV10N1\\_IQ-34.pdf](http://www.icmagazineonline.com/current/pdf/Pg28-29-IQV10N1_IQ-34.pdf) (23.02.2013),
- [18] HINSCH, Hermann: *Elektronik: Ein Werkzeug für Naturwissenschaftler*. [Nachdr.]. Berlin [u.a.] : Springer, 1996. – ISBN 3-540-61360-9
- [19] TIETZE, Ulrich ; GAMM, Eberhard ; SCHENK, Christoph: *Halbleiter-Schaltungstechnik*. 13. Heidelberg [u.a.] : Springer, 2010. – ISBN 978-3-642-01621-9
- [20] MANOLAKIS, Dimitris G. ; INGLE, Vinay K.: *Applied digital signal processing: Theory and practice*. Cambridge : Cambridge University Press, 2011. – ISBN 9780521110020
- [21] GRÜNIGEN, Daniel Ch v.: *Digitale Signalverarbeitung: Mit einer Einführung in die kontinuierlichen Signale und Systeme ; mit 91 Beispielen, 80 Aufgaben sowie einer CD-ROM mit Lösungen sowie Entwurfs- und Simulationsprogrammen*. 4. München : Fachbuchverl. Leipzig im Carl-Hanser-Verl., 2008. – ISBN 978-3-446-41463-1
- [22] MITRA, Sanjit K.: *Digital signal processing: A computer-based approach*. 3. Boston : McGraw-Hill, 2006. – ISBN 9780071244671
- [23] MERTINS, Alfred: *Signaltheorie: Grundlagen der Signalbeschreibung, Filterbänke, Wavelets, Zeit-Frequenz-Analyse, Parameter- und Signalschätzung ; mit 158 Abbildungen und 5 Tabellen*. 2. Wiesbaden : Vieweg + Teubner, 2010. – ISBN 3834807370

- 
- [24] USB IMPLEMENTERS FORUM, Inc.: *Universal Serial Bus Device Class Definition for Audio Devices*. [http://www.usb.org/developers/devclass\\_docs/audio10.pdf](http://www.usb.org/developers/devclass_docs/audio10.pdf) (22.03.2013),
- [25] Audacity®. <http://audacity.sourceforge.net/> (08.02.2013),
- [26] TCHEGHO, A. ; GRÄB, H. ; SCHLICHTMANN, U. ; MATTES, H. ; SATTLER, S.: *Analyse und Untersuchung der Quantisierungseffekte beim Goertzel-Filter*. <http://www.adv-radio-sci.net/7/73/2009/ars-7-73-2009.pdf> (22.03.2013),
- [27] SIGNAL RECOVERY: *Model 7265 DSP Lock-in Amplifier Instruction Manual*. <http://www.signalrecovery.com/download/190284-A-MNL-C.pdf> (08.02.2013),

# Anhang

Die folgende Liste erklärt die Ordnerstruktur der beigelegten CD.

## **PAS Messungen**

Alle Aufzeichnungen der Messungen sind in diesem Ordner zusammengefasst. In den nach dem Datum des Messtages benannten Unterordnern befinden sich jeweils aufgenommenen Messreihen. Die Benennung der Dateien wurde dabei so gewählt, dass die WAV Dateien *xx.wav* Daten enthalten, die mit dem MEMS-Mikrofon aufgenommen wurden und die Dateien *analog\_xx.wav* Daten enthalten, die mit dem Elektret-Mikrofon aufgenommen wurden. *xx* steht dabei für eine zweistellige Zahl, die bei 01 beginnt. Im Fall der Goertzel-Algorithmus Firmware sind die Ergebnisse in den Dateien *xx.txt* gespeichert worden.

In den Excel Dateien *Messreihe\_[DD]\_[MM]\_[YYYY].xls* wurde zu den fortlaufenden, zweistelligen *xx* Nummern der Aufzeichnungen immer jeweils die Eingangsleistung des Lasers, die optische Ausgangsleistung, die momentane Wellenlänge, der angezeigte Messwert des Lock-In Verstärkers, sowie der Etalon Winkel des optisch-parametrischen Oszillators, die Umgebungstemperatur nahe der Messzelle und die momentane Chopper-Frequenz aufgenommen.

In frühen Messungen wurden

## **Dokumente**

In diesem sind alle in dieser Arbeit genannten oder verwendeten Datenblätter und Handbücher zusammengetragen sowie der Konstruktionsplan der Messzelle.

## **Firmware**

In diesem Ordner befinden sich die beiden Unterordner *USB Mikrofon* und *Goertzel-Algorithmus*. In ihnen ist das jeweilige Code Composer Studio Softwareprojekt enthalten.

## **Matlab**

In diesem Ordner sind alle Matlab Skripte enthalten, die für die Anfertigung der Abbildungen und die Auswertungen der Messdaten erzeugt wurden.

# Versicherung über die Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit im Sinne der Prüfungsordnung nach §16(5) APSO-TI-BM ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen habe ich unter Angabe der Quellen kenntlich gemacht.

Hamburg, 25. April 2013

Ort, Datum

Unterschrift