

Laborversuch

Mittelungsfilter

Im ersten Teil dieses Laborversuchs soll der sich auf dem DSK (DSP Starter Kit) befindende Stereo-Codec (Coder/Decoder) TLV320AIC23B von TI näher betrachtet werden. Mit einem einfachen Programm sollen per Interrupt-Betrieb Abtastwerte von den beiden AD-Wandlern des Codec gelesen und anschließend wieder unverändert über die beiden DA-Wandler ausgegeben werden (sog. Talk-through-Betrieb). Dabei soll die maximal mögliche analoge Aussteuerung der AD-Wandler ermittelt werden.

Der zweite Teil des Laborversuchs verdeutlicht die Wirkungsweise eines zu programmierenden digitalen Mittelungsfilters. Mit Hilfe der Soundkarte im Labor-PC und einer entsprechenden Windows-Audio-Software soll die Übertragungsfunktion des Filters ermittelt werden.

1. Einführende Informationen

1.1 Quell-Code-Vorgaben

Wie bei den letzten beiden Laborversuchen, steht Ihnen auch dieses mal wieder, zumindest für den ersten Teil, ein vorbereitetes, lauffähiges Projekt zur Verfügung.

Um die im Projektordner *Project3* enthaltene Datei *codec_in_out.c* übersichtlicher zu gestalten, wurden die Funktionen zur Initialisierung der McBSPs und des Codecs in die separate Datei *init_mcbbsp_codec.c* ausgelagert und über die Header-Datei *init_mcbbsp_codec.h* in *codec_in_out.c* eingebunden.

Zusätzlich zum letzten Laborversuch wurde die Interrupt-Service-Routine *isr12_rint1* für den McBSP1-Receive-Interrupt in das Projekt aufgenommen und über das Interrupt-Enable-Register entsprechend freigegeben.

1.2 Die Windows-Audio-Software

Auf den Labor-PCs steht Ihnen das Programmpaket AudioTester zur Verfügung, das verschiedene Messfunktionen beinhaltet wie z.B. Sweep-Measurement, FFT, Oscilloscope.

Mittels Fast-Fourier-Transformation (FFT) können Spektraluntersuchungen durchgeführt werden. Mit dem Generatorteil der Messfunktion FFT lassen sich über die eingebaute Soundkarte des PC's u.a. sinusförmige Signale einstellbarer Frequenz und Amplitude erzeugen.

Mit der Oszilloskopfunktion kann ein am Line-Eingang der Soundkarte anliegendes Signal graphisch dargestellt werden.

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker

Und mit dem eingebauten Wobbelgenerator kann der Amplitudengang eines Übertragungssystems aufgenommen werden. Hierbei werden über den Line-Ausgang der Soundkarte Sinusschwingungen definierter Frequenz ausgegeben, über das zu untersuchende System übertragen (in unserem Fall das Signalverarbeitungssystem, z.B. mit implementiertem Mittelungsfiler) und über den Line-Eingang wieder der Soundkarte zugeführt. Auf diese Weise kann der Amplitudengang $|H(f)|$ des Systems analysiert und graphisch dargestellt werden.

2. Erster Teil: Talk-through-Betrieb

2.1 Vorbereitung und Durchführung

2.1 Durchführung:



Kopieren Sie sich den Projektordner **Project3.zip** von der Labor-Internetseite auf die lokale Festplatte in das Verzeichnis `c:\userdata\Ihr_Arbeitsverzeichnis\` bzw. in *Eigene Dateien* (siehe Einführungsdokument „Organisatorisches“) und **entpacken** Sie den Projektordner in das **Unterverzeichnis eclipseworkspace_v5_1** (vgl. dazu auch LV1, Teil 1, Kapitel 2.1). Die Labor-Internet-Seite erreichen Sie auf den Labor-PCs im Raum T1.4.01 am schnellsten über das Startmenü:


Start -> Labore -> Signale und Systeme -> Labor Signale und Systeme

Versorgen Sie das Demoboard (DSK6713) mit Spannung (Netzteil) und warten Sie den jetzt ablaufenden PowerOnSelfTest (POST) ab. Starten Sie erst danach die Entwicklungsumgebung Code Composer Studio v5 über das Startmenü:

Start -> Labore -> Signale und Systeme -> Code Composer Studio

2.2 Durchführung:

Um nicht mit einem anderen Projekt aus den vorangegangenen Laborversuchen in Konflikt zu geraten, sollten diese mit Hilfe des zugehörigen Kontextmenüs geschlossen werden (sofern dies noch nicht geschehen ist). Klicken Sie dazu mit der rechten Maustaste auf das jeweilige  Project im  Project Explorer und wählen Sie den Eintrag Close Project.

Importieren Sie das dritte Projekt indem Sie im Menü **Project** den Eintrag  Import **E_xisting CCS/CCE Eclipse Project** wählen (vgl. LV1, Teil 1, Kapitel 2.3).

Verschaffen Sie sich einen Überblick über die in dieses Projekt eingebundenen Dateien. (Erläuterungen siehe Infoblatt „Code Composer Studio...“)

2.3 Vorbereitung und Durchführung:

Sehen Sie sich die Dateien **codec_in_out.c** und **vectors3.asm** genauer an und versuchen Sie die Funktionsweise nachzuvollziehen !!!

Die Übergabe der Abtastwerte (Samples) von der Receive-Interrupt-Routine an die Transmit-Interrupt-Routine läuft über den global angelegten **Ringspeicher** `data_buffer` der Länge `L`. Dadurch ist es zum einen leicht möglich, sich einen Zeitausschnitt gewünschter Länge `L` (bei einer Abtastrate des Codecs von 32 kHz also `L · 31,25 μs`) des digitalgewandelten Signals über die Graph-Funktion von CCS anzusehen, zum anderen kann dadurch im zweiten Teil des Laborversuchs die gewünschte Mittelungslänge leicht ein- bzw. umgestellt werden.

Die Verwaltung des ebenfalls global angelegten Zeigers `j` für den Zugriff der Interrupt-Service-Routinen auf den Ringspeicher erfolgt in der Receive-Interrupt-Routine.

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker

2.4 Durchführung:

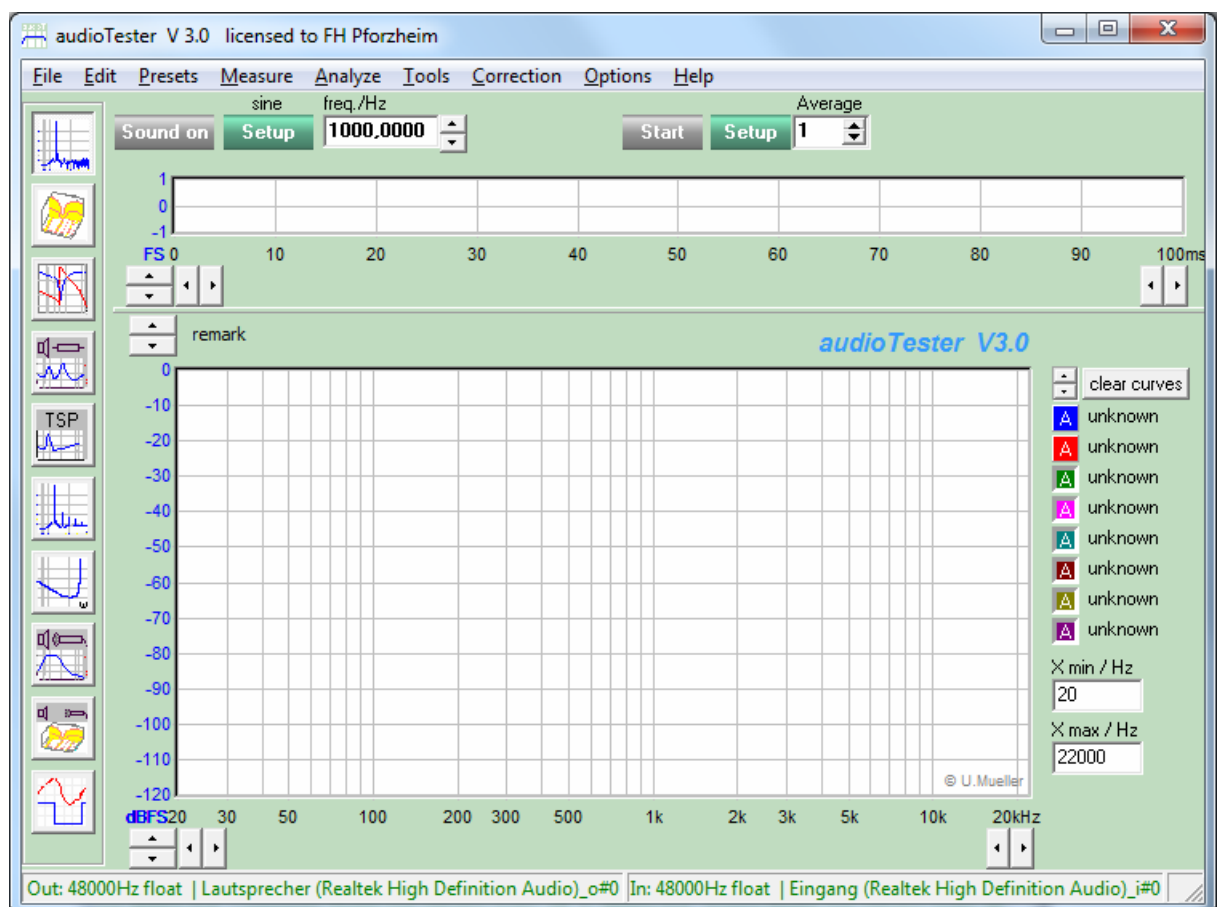
Stellen Sie in der Entwicklungsumgebung Code Composer Studio v5 durch Starten des Debuggers eine software-mäßige Verbindung zum Demoboard her. Dadurch wird gleichzeitig die Objekt-Datei *Project3.out* auf das DSK6713 heruntergeladen.

Starten Sie das DSP-Programm.

Starten Sie ebenfalls das Software-Paket AudioTester:

Start -> Labore -> Signale und Systeme -> audioTester

In der ersten Messfunktion FFT von AudioTester (2D FFT, 2D Spektralanalyse, erstes Symbol in der Symbolleiste) ist ein **Wave-Generator** integriert bzw. erreichbar, mit dem Sie den Durchreichebetrieb zunächst testen sollen.

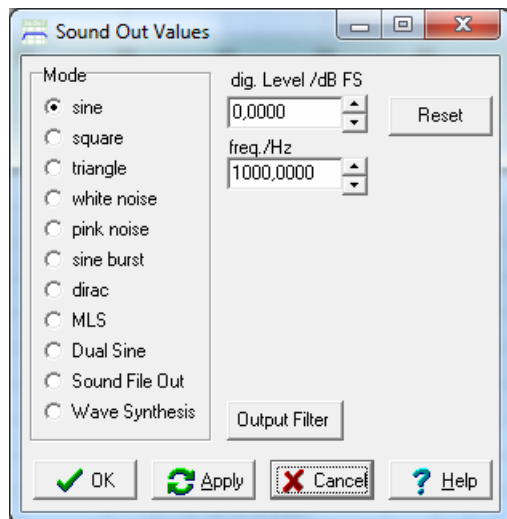


Stellen Sie dazu unter Windows über das Lautsprechersymbol rechts unten in der Taskleiste einen Ihnen als sinnvoll erscheinenden aber nicht zu hohen, analogen Ausgangspegel für die Tonwiedergabe ein. Der digitale Pegel des Wave-Generators in AudioTester sollte 0 dBFS betragen (linke Schaltfläche „Setup“). Stellen Sie die Signalform auf Sinus und eine Frequenz von z.B. 1000 Hz ein.

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker



Schalten Sie danach die Signalausgabe durch einen Klick auf „Sound on“ ein.


- 2.5 Hören Sie sich den vom Codec des DSK wieder ausgegebenen Sinuston über den Kopfhörer an. Es sollte ein sauberer Sinuston zu hören sein. (Achtung: Einstellungen am Mischpult überprüfen!)



Schalten Sie den Abhörpunkt mit Hilfe des TAPE TO PHONES - Schalters am Mischpult hin und her und erhöhen Sie dabei den Ausgangspegel des Wave-Generators. In Abhängigkeit vom tatsächlichen analogen Ausgangspegel der im PC verbauten Soundkarte und der Empfindlichkeitseinstellung der Analogeingänge des Codecs (vgl. `init_mcbasp_codec.c`) ist dabei auch eine Übersteuerung der AD-Wandler im Codec des Demoboards möglich. Wenn jetzt also das Ausgangssignal des Signalverarbeitungssystems hörbare Verzerrungen aufweist, dann haben Sie am Wave-Generator bzw. unter Windows einen zu hohen Pegel eingestellt. Reduzieren Sie in diesem Fall den Pegel wieder, bis ein sauberer Sinuston zu hören ist.

- 2.6 Stoppen Sie das DSP-Programm und lassen Sie sich den letzten Inhalt des Ringspeichers `data_buffer` (von vor dem Stoppen) graphisch darstellen.

Öffnen Sie dazu über das Menü **Tools** -> **Graph** -> **Single Time** das Graph-Properties-Fenster. Dort müssen Sie unter Data Properties die Startadresse bzw. den der Startadresse zugeordneten Namen des darzustellenden Speicherbereichs, also `data_buffer`, angeben.

Weiter muss noch die Anzahl der aus dem Speicher zu holenden Daten (Wert von `L`) und deren Interpretation angegeben werden (Achtung: in einem 32 Bit Datenwort sind zwei 16 Bit Samples im K2-Format enthalten – Sie müssen deshalb für eine korrekte Darstellung hier noch eine weitere Einstellung anpassen! Welche?).

Nach der graphischen Darstellung erreichen Sie das Graph-Property-Dialog-Fenster übrigens auch über das Symbol  in der Symbolleiste des Fensters.

Wenn Sie wiederholt die Graph Properties anpassen, wird jedes mal ein neuer Datensatz aus dem Speicher geholt und ans Ende der Darstellung angehängt. Lassen Sie sich dadurch nicht irritieren und löschen Sie ggf. die komplette Darstellung über das Symbol  in der Symbolleiste des Fensters und lassen Sie über das Symbol  einen neuen Datensatz aus dem Speicher holen und darstellen.

Graphische Eigenschaften ändern:

Über das zur graphischen Darstellung gehörende Kontextmenü können einige graphische

Eigenschaften der Darstellung beeinflusst werden (mit der rechten Maustaste in die Grafik klicken und im sich öffnenden Kontextmenü den gewünschten Eintrag wählen).


Dort kann unter Display as beispielsweise die hier sinnvolle Linienart Connected Line with markers gewählt werden (das geht übrigens auch über den Eintrag Display Properties... und dort in der Rubrik General mit Hilfe von Display data as).


Weiter sollte über den Eintrag Display Properties... (im Kontextmenü ganz unten) und dort in der Rubrik Axes unter Y-Axis das Display format General gewählt sein bzw. werden, woraus eine ganzzahlige Y-Achsenbeschriftung resultiert (passend zum Datenformat Signed Integer).

- 2.7 Optimieren Sie nun mit Hilfe der graphischen Darstellung die Aussteuerung der A/D-Wandler im Codec über die Pegeleinstellung des Wave-Generators so, dass der Wertebereich des ADC (digitaler Dynamikbereich) möglichst gut ausgenutzt wird, es aber zu keiner Übersteuerung kommt.

Je nach im PC verbauter Soundkarte ist hier wegen möglicherweise zu geringem maximalem analogem Ausgangspegel nur ein suboptimales Ergebnis erzielbar (Aussteuergrenze der A/D-Wandler im Codec kann ggf. nicht erreicht werden).

Dabei ist die Messfunktion der graphischen Darstellung hilfreich:




Über das Symbol  in der Symbolleiste des Fensters lässt sich der Measurement Marker Mode umschalten, wobei der eine der beiden Modi ein einfaches Ausgeben der X- und Y-Werte des dargestellten Kurvenzugs an der aktuellen Mauszeigerposition im linken oberen Fensterbereich zur Verfügung stellt (dabei sollte der Mauszeiger grob entlang des Kurvenzugs bewegt und nicht in die Darstellung geklickt werden! Ein Klicken bewirkt das Setzen eines Measurement Markers).

Über das Symbol  rechts daneben lassen sich die Eigenschaften der Messfunktion festlegen (sinnvoll: • Snap to Data, • Both).

- 2.8 Nach der korrekten Pegeleinstellung sollten Sie sich die Abtastwerte auch einmal direkt als Hex-Zahlenwerte im Ringspeicher ansehen. Öffnen Sie dazu über **V**iew -> **M**emory **B**rowser das entsprechende Anzeigefenster und geben Sie dort ebenfalls die Startadresse des Ringspeichers, also `data_buffer`, an.

Sehen Sie sich die Ausgabe an und wechseln Sie anschließend die Interpretation der ausgegebenen Daten von 32 Bit Hex auf 16 Bit Hex und vergleichen Sie die beiden Ausgaben (die 16 Bit Hex-Darstellung zeigt natürlich zuerst den niederwertigen 16 Bit-Wert der 32 Bit Hex Darstellung). Können Sie der Ausgabe den sinusförmigen Werteverlauf ansehen? Wechseln Sie abschließend die Interpretation auf 16 Bit Signed Integer. Die Adressangaben in der linken Spalte sind übrigens Byte-Angaben (zählen Sie ruhig einmal durch und überprüfen Sie, ob es sich tatsächlich um eine Byte-Angabe handelt).

- 2.9 In der Transmit-Interrupt-Routine wurde eine zusätzliche **Bit-Masken-Variable** eingeführt, mit der Sie die 16-Bit-Auflösung der auszugebenden Samples aus dem Ringspeicher bis hinunter auf 1 Bit reduzieren können.

Übernehmen Sie die Variable `mask` in die Expressions View (Vorgehensweise wie im letzten Laborversuch in Kap. 2.2 beschrieben entweder direkt im Fenster  Expressions über das Symbol  oder im  C-Quellcodefenster durch markieren der Variable `mask` und öffnen des zugehörigen Kontextmenüs) und ändern Sie den Vorgabewert von `0xFFFFFFFF` auf z.B. `0xFF00FF00` für eine 8-Bit-Auflösung ab.

Dazu müssen Sie die Interpretation des Wertes in der Expressions View zuvor auf Hex umstellen. Markieren Sie deshalb in der Expressions View zunächst die Zeile mit der Variable bzw. Expression `mask` und öffnen Sie mit Hilfe der rechten Maustaste das zugehörige Kontextmenü. Darin wählen Sie den Eintrag Number Format, stellen auf Hex

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker

um und ändern anschließend den Wert der Variable direkt in der Spalte Value auf 0xFF00FF00 ab.

Starten Sie das DSP-Programm und hören Sie sich den vom Codec des Demoboards ausgegebenen und jetzt in der Auflösung reduzierten Sinuston wieder über den Kopfhörer an. Die Auflösungsreduktion macht sich durch rauschähnliche Störgeräusche bemerkbar. Testen Sie daraufhin schrittweise wie sich eine weitere Reduzierung der Auflösung auf zunächst 4 Bit und anschließend bis hinunter auf 1 Bit (entspricht der Ausgabe von 0 oder -32768) auswirkt.

Geben Sie mit Hilfe des Windows Media Players anstelle des Sinustons auch eine geeignete Sprach- oder Musik-Datei über das DSK wieder und vergleichen Sie den Höreindruck der unterschiedlichen Auflösungen.

(Über das Windows-Startmenü finden Sie unter Vorlagen und dort im Signale und Systeme - Verzeichnis ebenfalls Sprach- und Musikdateien, beispielsweise bbauer16.wav, tagesschauohne.wav, ...)

Es ist überraschend, dass selbst bei Wiedergabe mit nur einem Bit Auflösung noch ausreichend Informationsgehalt vorhanden ist, beispielsweise bei einer Sprachdatei, um Information entnehmen zu können.

2.2 (Vorbereitungs-) Fragen

- F 2.1 Was müssen Sie bei der Graph-Funktion von CCS für den Parameter `DSP Data Type` angeben und welchen weiteren Parameter müssen Sie ändern, um eine sinnvolle Darstellung der Abtastwerte im Ringspeicher zu erhalten?
Erläutern Sie kurz wie Sie zu Ihrer Antwort kommen.
- F 2.2 Warum weist der über die Graph-Funktion von CCS dargestellte Signalverlauf (Speicherabbild der Abtastwerte im Ringspeicher) aller Wahrscheinlichkeit nach eine Sprungstelle auf?
- F 2.3 Zwischen Ein- und Ausgangssignal des Codecs besteht eine Verzögerung (ein Delay) von $L \cdot 31,25 \mu\text{s}$ (zusätzlich zum System-Delay). Warum?
Erläutern Sie den Ablauf!
- F 2.4 Welchen Inhalt muss die Variable `mask` für eine Auflösungsreduzierung auf 3 Bit, 2 Bit und 1 Bit jeweils erhalten?

Die Fragen sind handschriftlich in der Versuchsvorbereitung bzw. Ihrem Versuchsprotokoll zu beantworten!

3. Zweiter Teil: Mittelwertbilder

3.1 Theorie

In diesem Versuchsteil soll ein einfaches digitales **Tiefpaßfilter**, bestehend aus einem gleitenden Mittelwertbilder programmiert werden. Das Filter addiert hierbei die letzten L Abtastwerte und dividiert anschließend durch L . In Abbildung 3.1 ist dieses Filter dargestellt. Es besteht aus einer sog. "Tapped-Delay-Line", d.h., einer Kettenschaltung von Schieberegister-speichern, die das Eingangssignal um jeweils einen Abtastwert $T = 1/f_s$ verzögern.

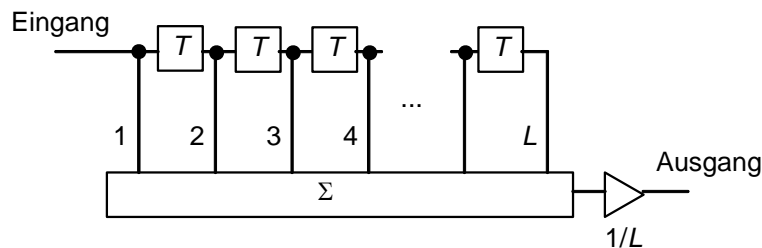


Abbildung 3.1: Mittelungsfilter (gleitender Mittelwertbilder)

Die **Impulsantwort** $h(k)$ des Filters ist

$$h(k) = \frac{1}{L} \sum_{i=0}^{L-1} \delta(k-i) \quad (3.1)$$

Abbildung 3.2a) zeigt $h(k)$; es handelt sich also um eine Folge von L Dirac-Impulsen mit dem jeweiligen Gewicht $1/L$. Zur Ermittlung der **Übertragungsfunktion** $H(\Omega)$ des Filters wird $h(k)$ zunächst Z-transformiert. Wir erhalten

$$H(z) = \frac{1}{L} \sum_{i=0}^{L-1} z^{-i} = \frac{1}{L} \cdot \frac{1-z^{-L}}{1-z^{-1}} \quad (3.2)$$

Mit $z = e^{j\Omega}$ ergibt sich aus Gl.(3.2) der **Betragsfrequenzgang** des Filters zu

$$|H(\Omega)| = \frac{1}{L} \cdot \frac{|1-e^{-jL\Omega}|}{|1-e^{-j\Omega}|} = \frac{1}{L} \cdot \sqrt{\frac{(1-\cos L\Omega)^2 + (\sin L\Omega)^2}{(1-\cos \Omega)^2 + \sin^2 \Omega}} \quad (3.3)$$

mit $\Omega = 2\pi fT$.

Abbildung 3.2b) zeigt den prinzipiellen Verlauf des Frequenzgangs nach Gl.(3.3).

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker

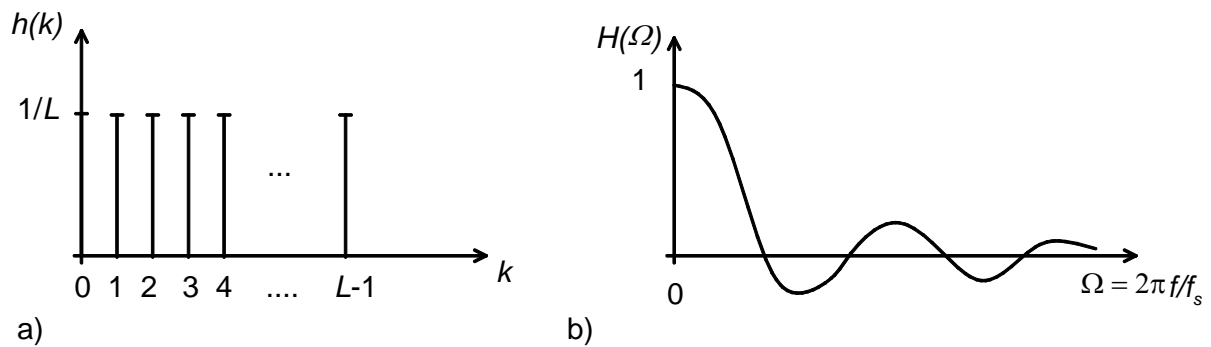


Abbildung 3.2: Impulsantwort $h(k)$ (a) und Frequenzgang $H(\Omega)$ (b)

Abbildung 3.3 zeigt den Betragsfrequenzgang nach Gl.(3.3) in logarithmischer Darstellung für $L = 10$. Man erkennt den Tiefpaß-Charakter des Filters. Die Grenzfrequenz f_g hängt von der Mittelungslänge L und der Abtastfrequenz f_s ab.

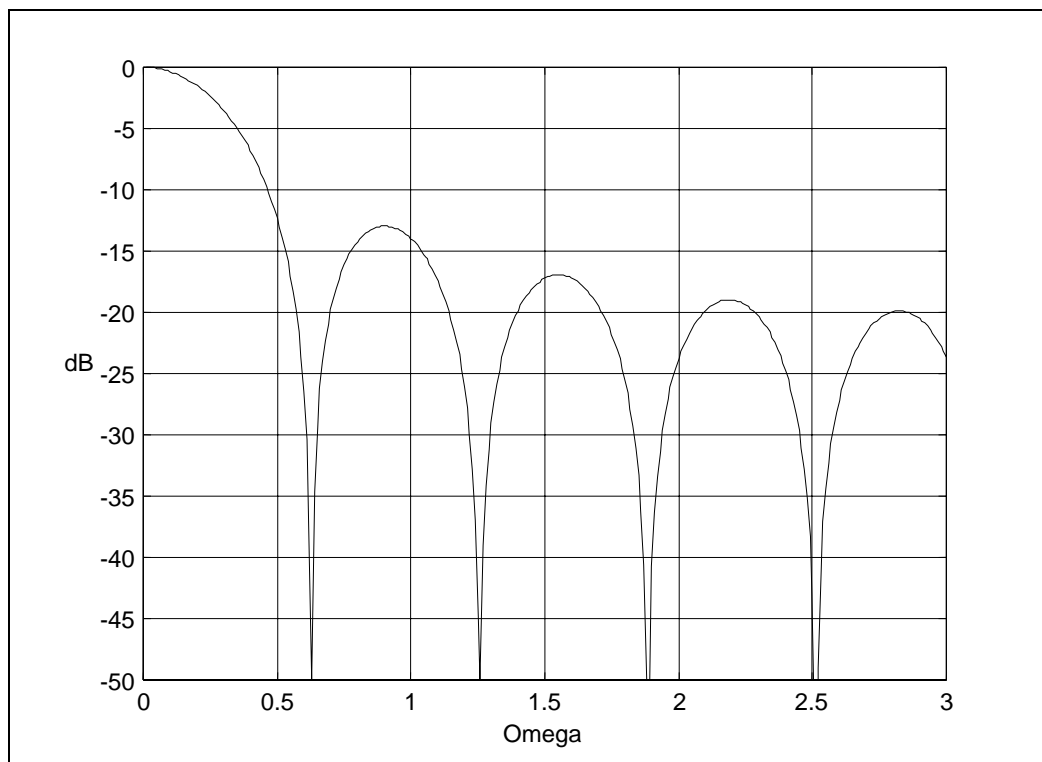


Abbildung 3.3: Betragsfrequenzgang eines Mittelungsfilters mit $L = 10$

3.2 Vorbereitung und Durchführung

3.1 Vorbereitung:

Die Abtastfrequenz der ADCs im Codec beträgt $f_s = 32$ kHz. Bestimmen Sie über Gl.(3.3) die Mittelungslänge L für eine erste Nullstelle des Frequenzgangs bei $f_0 = 800$ Hz. Bei welchen Frequenzen treten weitere Nullstellen auf?


SS 2013

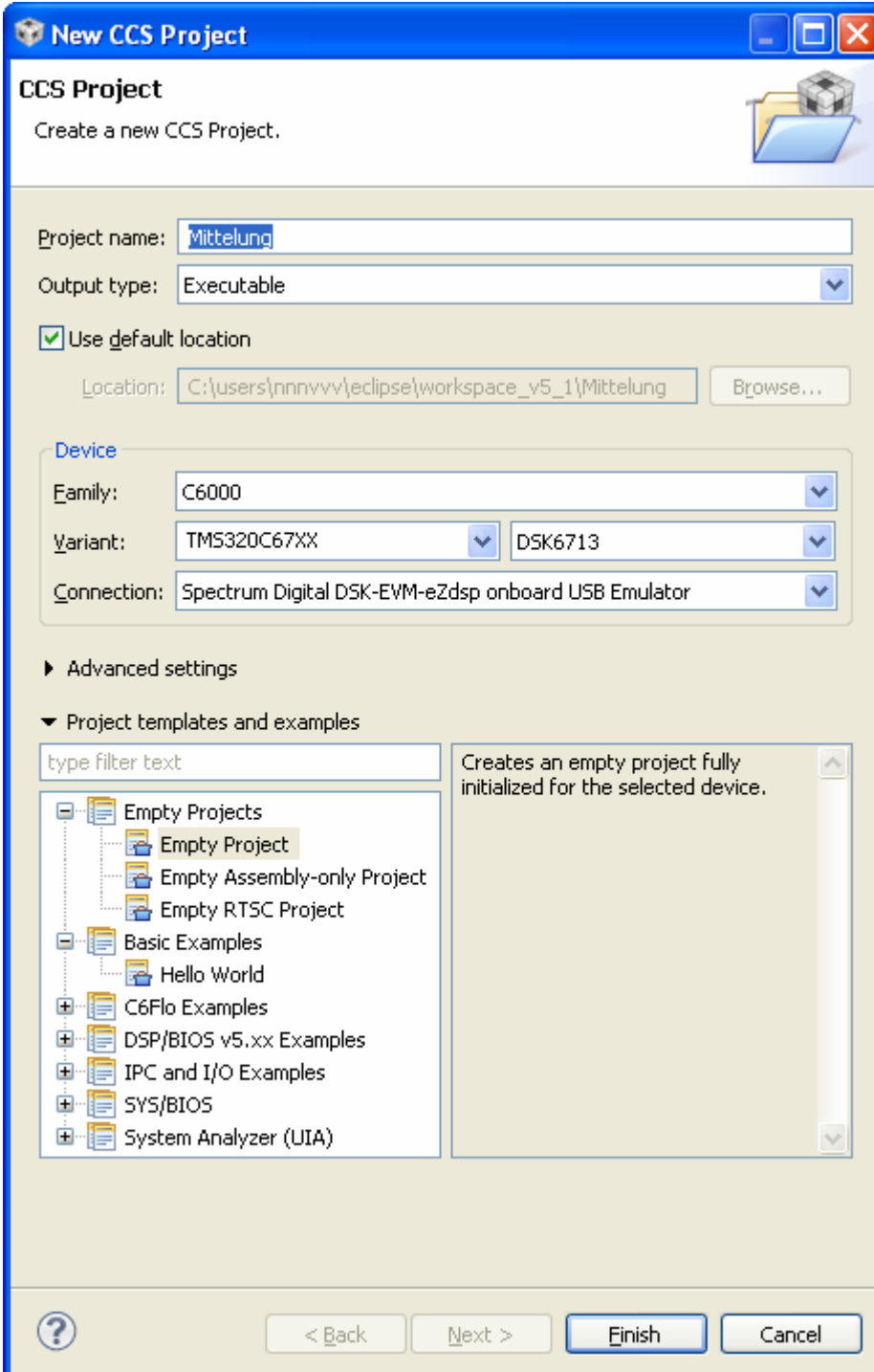
Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker

3.2 Durchführung:

Neues Projekt erstellen

Erweitern Sie das Programm aus Teil 1 um den Mittelwertbilder. Erzeugen Sie hierzu über **Project** ->  **New CCS Project** ein neues Standard-C-Projekt für das Demoboard DSK6713. Geben Sie dem Projekt einen neuen Namen, beispielsweise **Mittelung** und wählen Sie im Bereich **Device** der Reihe nach die in der nachfolgenden Abbildung zu sehenden DSK-spezifischen Einstellungen. Bestätigen Sie danach mit **Finish**.



New CCS Project

CCS Project
Create a new CCS Project.

Project name:

Output type:

☒ Use default location

Location:

Device

Family:

Variant:

Connection:

► Advanced settings

▼ Project templates and examples

type filter text


- Empty Projects
 - Empty Project
 - Empty Assembly-only Project
 - Empty RTSC Project
- Basic Examples
 - Hello World
- C6Flo Examples
- DSP/BIOS v5.xx Examples
- IPC and I/O Examples
- SYS/BIOS
- System Analyzer (UIA)


Creates an empty project fully initialized for the selected device.



SS 2013

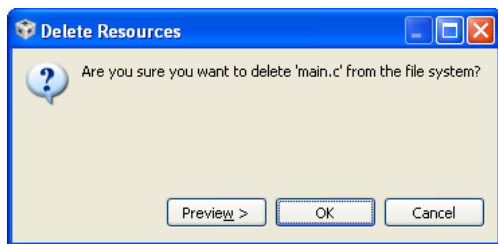
Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker

Das neu erstellte Projekt  **Mittelung** enthält jetzt automatisch auch eine leere C-Quellcodedatei, die mit C-Code gefüllt werden kann. Da für diesen Laborversuch allerdings vorgefertigte Quelldateien zur Verfügung stehen, kann die C-Quellcodedatei *main.c* aus dem Projekt gelöscht werden.

Öffnen Sie dazu im Fensterbereich  **Project Explorer** das soeben angelegte Projekt  **Mittelung** (Mausklick auf  oder Doppelklick auf den Projektnamen), so dass Sie die zugehörigen Quelldateien angezeigt bekommen.



Löschen Sie die Datei *main.c* durch Auswahl der Funktion  **Delete** im Kontextmenü (Klick mit der rechten Maustaste auf *main.c* und im sich öffnenden Kontextmenü  **Delete** auswählen).

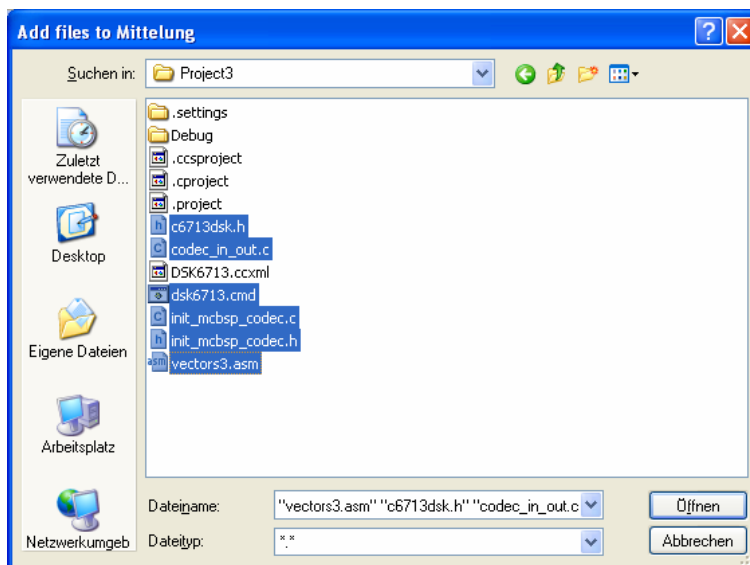


Bestätigen Sie die Löschanfrage mit **OK**.

Quelldateien dem Projekt hinzufügen

Fügen Sie dem neuen Projekt **Mittelung** die folgenden Quelldateien aus dem **Project3** hinzu: *c6713dsk.h*, *codec_in_out.c*, *dsk6713.cmd*, *init_mcbasp_codec.c*, *init_mcbasp_codec.h* und *vectors3.asm*

Wählen Sie dazu im  **Project Explorer** erneut das neue Projekt  **Mittelung** aus (durch markieren) und wählen Sie entweder im Menü **Project** den Eintrag **Add Files...** oder öffnen Sie mit der rechten Maustaste das zugehörige Kontextmenü und wählen darin **Add Files...** Danach das Verzeichnis **Project3** im Eclipse-Workspace-Ordner auswählen und darin die gewünschten Dateien markieren, die dem Projekt hinzugefügt werden sollen.

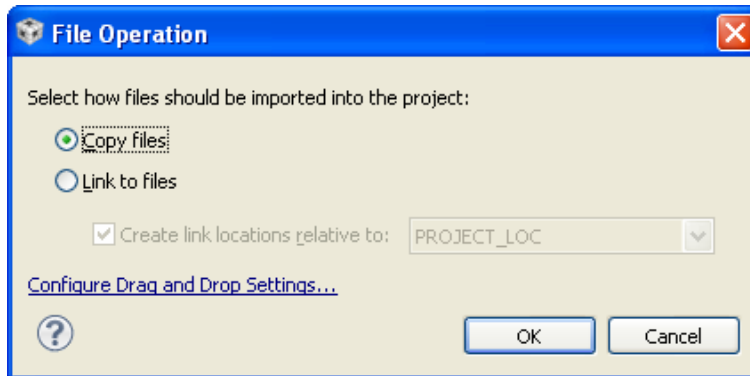


Die daraufhin folgende Kopierabfrage bestätigen Sie mit **OK**.



SS 2013

Fakultät für Technik, Studiengänge EIT/TI


Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker

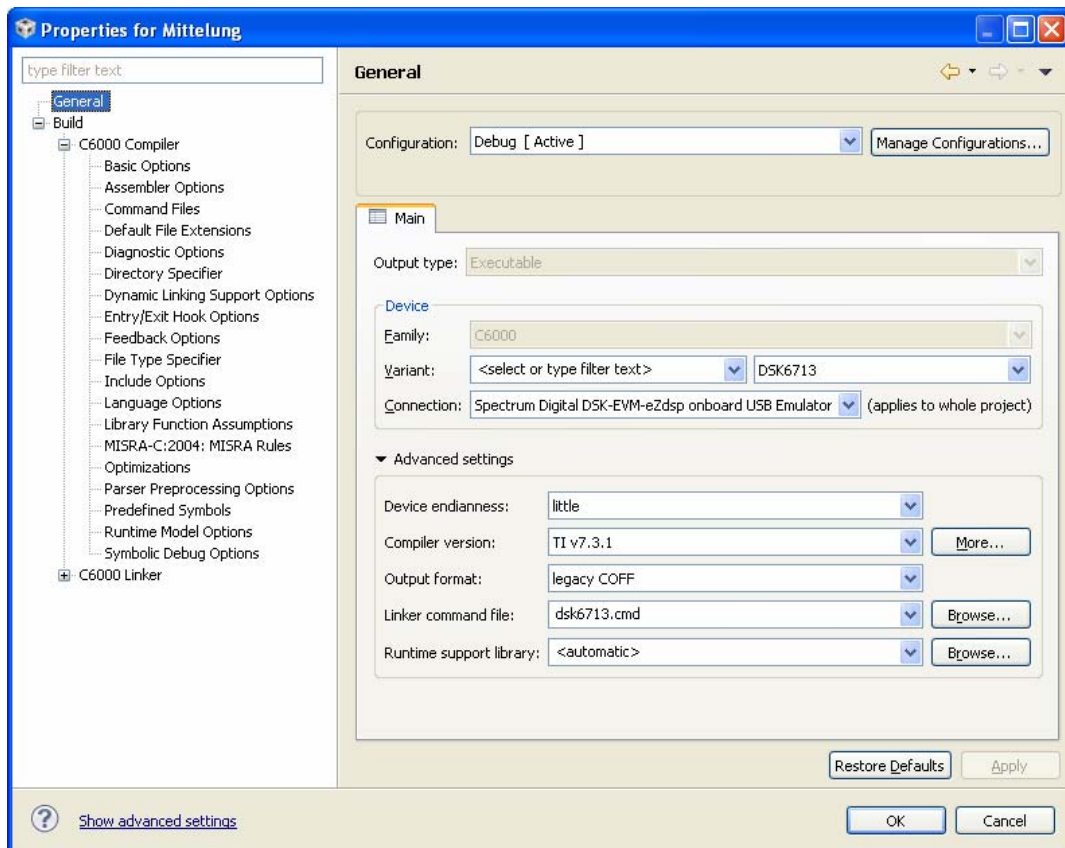


C-Quellcodedatei umbenennen

Markieren Sie jetzt im  Project Explorer die C-Quellcodedatei `codec_in_out.c` im Projekt  **Mittlung** und benennen Sie die Datei über das zugehörige Kontextmenü (rechte Maustaste -> Rename...) in **mittelung.c** um.

Projekteinstellungen überprüfen und ergänzen

Bei ausgewähltem Projekt im  Project Explorer entweder im Menü **Project** den Eintrag **Build Options...** wählen oder mit der rechten Maustaste das zugehörige Kontextmenü öffnen und darin **Build Options...** wählen.

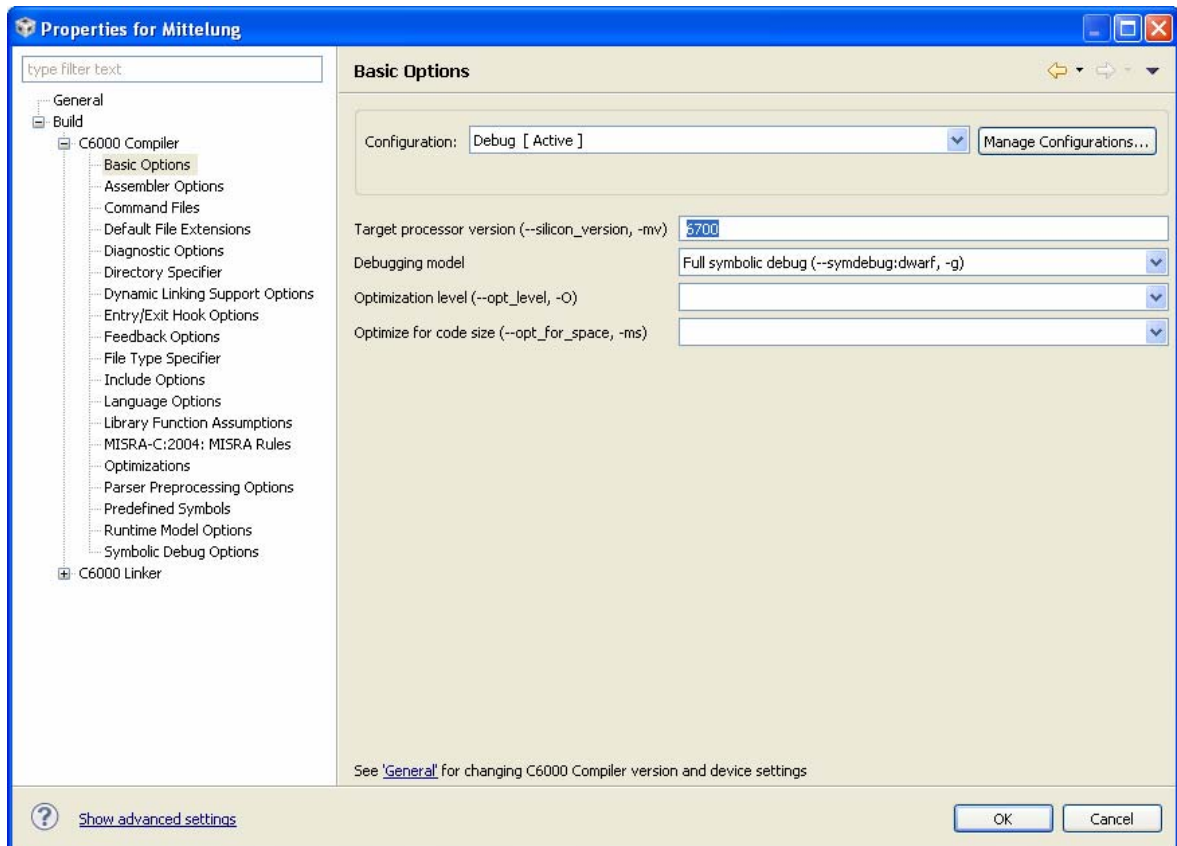


Spezifizieren Sie in der Projekt-Einstellung Build -> C6000 Compiler -> Basic Options die Option Target processor version **6700**.

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker



Im Linker-Bereich kann die Größe für den Heap und den Stack vorgeben werden, falls dies einmal notwendig werden sollte:

Build -> C6000 Linker -> Basic Options

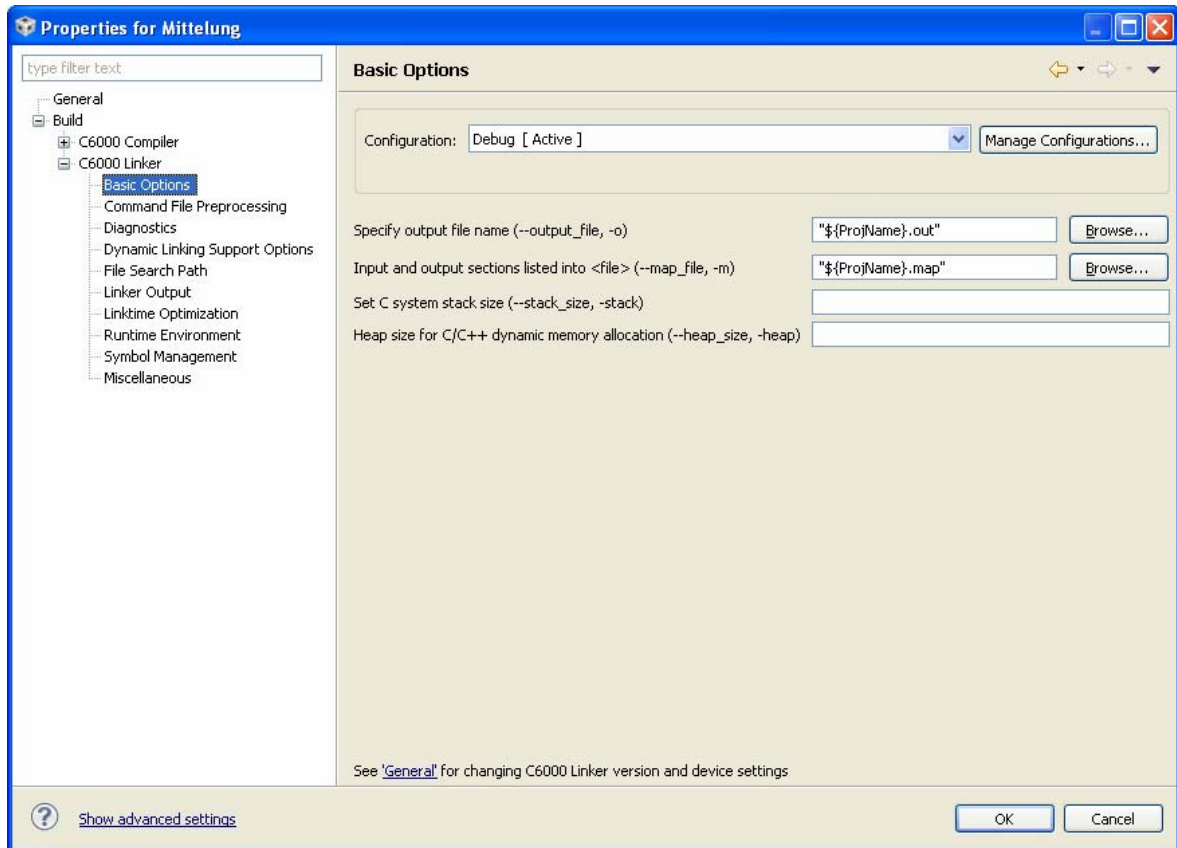
z.B. Stack Size: 1000
Heap Size: 1000

Ohne einen Eintrag wird die Größe auf jeweils 400 hex festgelegt (vgl. Meldung im Fenster Console nach einem Build-Prozess).

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker



3.3 Vorbereitung und Durchführung:

C-Code für die Filterberechnung erstellen

Erweitern Sie die C-Quelldatei *mittelung.c* um die **Mittelungsfilterberechnung** nach Abbildung 3.1, wobei die Tapped-Delay-Line durch den Ringspeicher realisiert ist.

Die Berechnung soll in der Endlosschleife von `main()` durchgeführt und durch Ändern eines Freigabesteuerwortes (eine global deklarierte Variable) in der Receive-Interrupt-Routine angestoßen werden. In der Receive-Interrupt-Routine also beispielsweise durch `start_calc = 1;`

Der Berechnungsstart in `main()` erfolgt dann durch eine if-Abfrage auf diese Variable mit anschließendem Rücksetzen von `start_calc`.

In der Transmit-Interrupt-Routine müssen Sie dann anstelle eines Datenwortes aus dem Ringspeicher das Ergebnis der Filterberechnung an den Codec ausgeben, beispielsweise über die global deklarierte Variable `unsigned int data_word;`

Überlegen Sie sich bereits vor Versuchsbeginn wie Sie vorgehen bzw. was Sie an welcher Stelle im C-Code wie programmieren müssen, um die gewünschte Funktionsweise zu erhalten!

Beachten Sie dabei, dass die Datenworte im Ringspeicher zwei 16 Bit Samples enthalten (linken und rechten Stereokanal).

Hinweise zu einem schrittweisen Vorgehen:

Der Einfachheit halber macht es Sinn, auf den Stereobetrieb zunächst zu verzichten und nur einen Kanal aus dem Ringspeicher zu filtern und an einen oder beide D/A-Wandler im Codec auszugeben. Die Ausgabe kann dann beispielsweise durch die folgende Zuweisung erfolgen, wenn die lokale 32 Bit Integer-Variable `sum` das Filterergebnis eines Kanals

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker

enthält:

```
data_word = (sum << 16) | (sum & 0x0000FFFF);
```

Für einen einfacheren Zugriff auf die 16 Bit Samples des linken oder rechten Stereokanals im 32 Bit breiten Ringspeicher können entsprechende Zeiger lokal in `main()` angelegt werden:




```
short *r_sample_buffer = (short *)data_buffer;  
short *l_sample_buffer = r_sample_buffer+1;
```

Der getrennte Zugriff auf die Samples über die Zeiger kann bzw. sollte in einem ersten Schritt mit Hilfe eines Durchreichtests in `main()` überprüft werden (also zunächst nur durchreichen, noch nicht filtern).

Der Zugriff bei einem Durchreichtest in `main()` ändert sich beispielsweise von:

```
data_word = data_buffer[j];    // durchreichen beider samples  
auf:
```

```
data_word = (((int)l_sample_buffer[2*j]) << 16) | (((int)r_sample_buffer[2*j] & 0x0000FFFF);
```

Programmieren Sie die Mittelungsfilterberechnung wie oben beschrieben und starten Sie anschließend Compiler und Linker indem Sie bei ausgewähltem/markiertem Projekt  Mittelung im  Project Explorer entweder im Menü Project den Eintrag  Build All wählen oder mit der rechten Maustaste das zugehörige Kontextmenü öffnen und darin Build Project wählen.

Nach erfolgreichem Build-Prozess erhalten Sie im Konsolenfenster die abschließende Meldung:

```
Finished building target: Mittelung.out
```

```
**** Build Finished ****
```

Starten Sie den Debugger und anschließend Ihr DSP-Programm und überprüfen Sie das Ergebnis wieder mit dem Kopfhörer. Verändern Sie dazu die Frequenz des Sinusgenerators und überprüfen Sie dabei die sich ergebende Dämpfung insbesondere bei den Nullstellen des Filters. Kontrollieren Sie die genaue Lage der Nullstellen mit den von Ihnen unter Aufgabenpunkt 3.1 berechneten Werten.

3.4 Durchführung:

Genaue Überprüfung durch Messung

Ermitteln Sie den Amplitudengang Ihres Filters mit Hilfe der Wobbel-Funktion (Sweep-Measurement) des Messprogramms audioTester.

Starten Sie die Messsoftware über das Startmenü des PC's

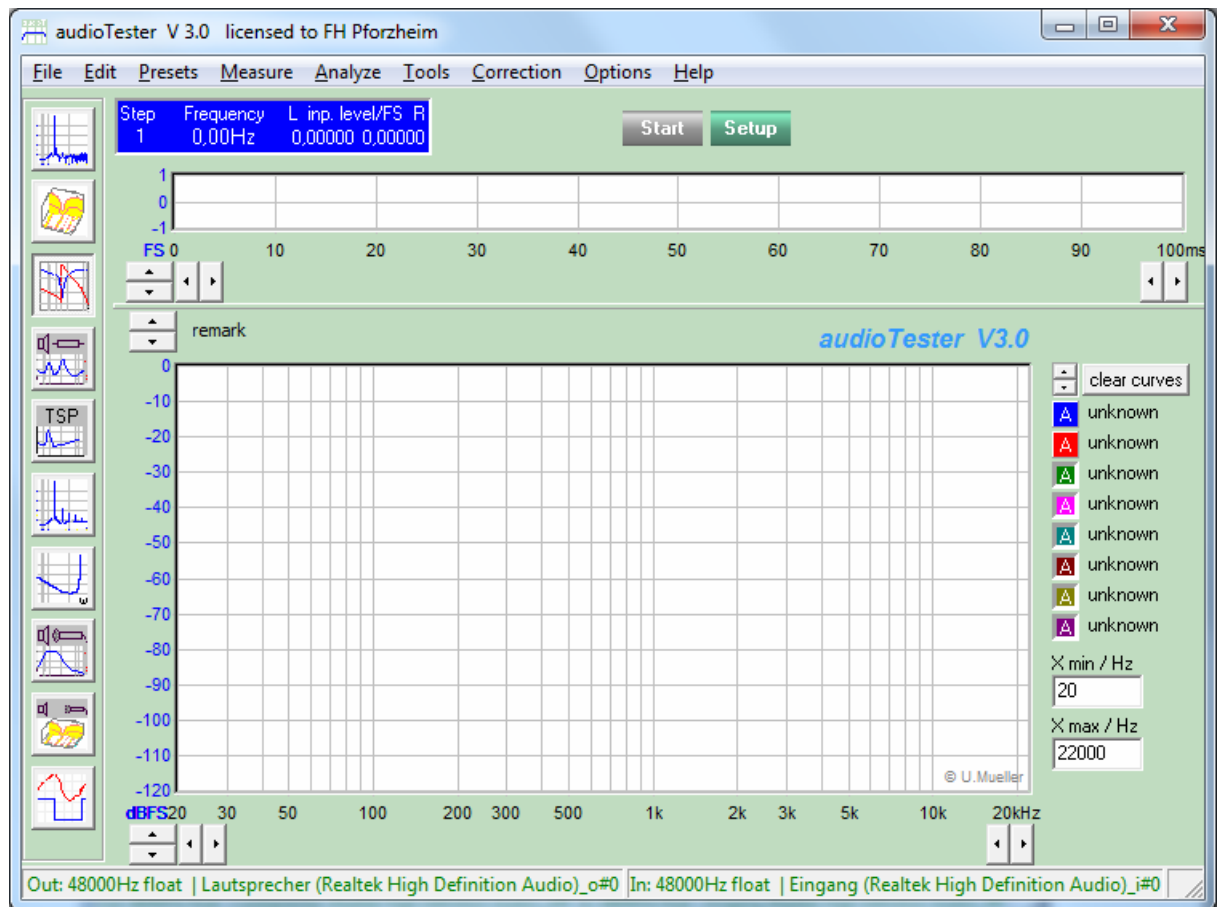
Start -> Labore -> Signale und Systeme -> audioTester

und stellen Sie über das dritte Icon der Symbolleiste die Meßmethode „Sweep Measurement“ ein.

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker



Pegelanpassung über die Windows Lautstärkeinstellungen:

Sie müssen den analogen Ausgangspegel des Sweep-Generators und die Eingangsempfindlichkeit der Soundkarte für die Messung über die Windows Lautstärkeinstellungen vornehmen. Diese erreichen Sie über das Lautsprechersymbol rechts unten in der Windows-Taskleiste.

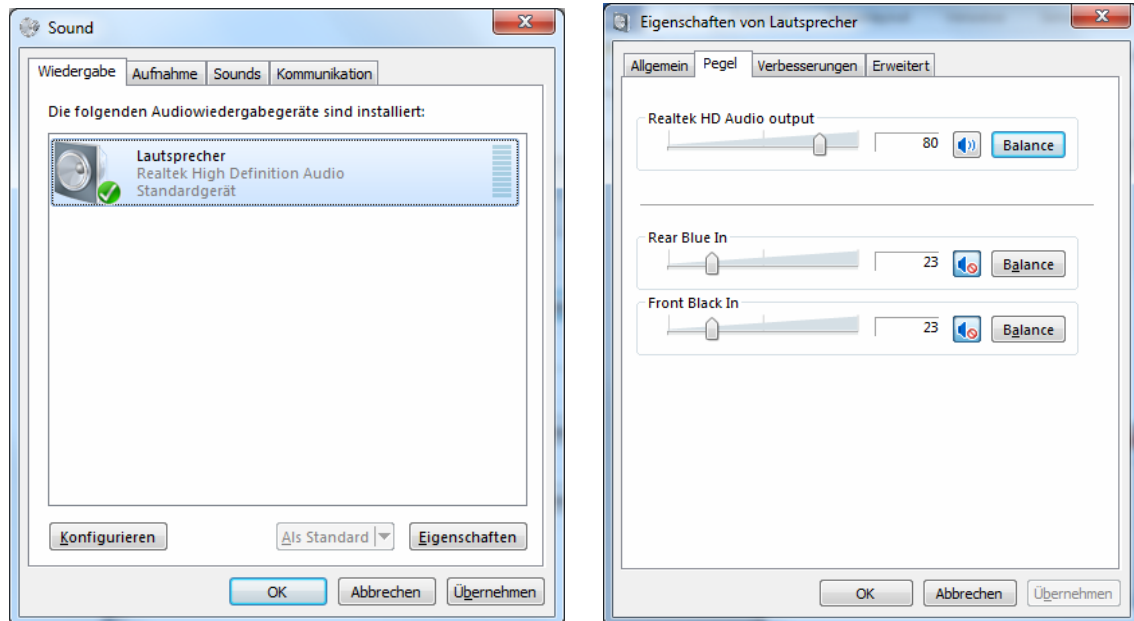
Den sinnvollen (maximalen) Ausgangspegel haben Sie im ersten Versuchsteil bereits ermittelt, für die Eingangsempfindlichkeit ist i.d.R. ein Wert im Bereich von 15 % bis 20 % sinnvoll.

Öffnen Sie zunächst die Lautstärkeinstellung für die Wiedergabe über einen Rechtsklick auf das Lautsprechersymbol. Im sich öffnenden Fenster können Sie über den Button Eigenschaften u.a. den Ausgangspegel des Messsignals einstellen.

SS 2013

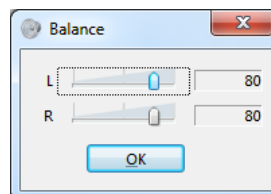
Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker



Vergewissern Sie sich, dass bei Ihnen – wie hier im rechten Fenster – die Wiedergabe des gemessenen Signals abgeschaltet ist (über Lautsprechersymbol für Rear Blue In). Sonst wird eine (Rückkoppel-)Schleife gebildet!

Über den Button Balance kann hier übrigens später die Links-/Rechts-Trennung der Stereoübertragung überprüft werden:

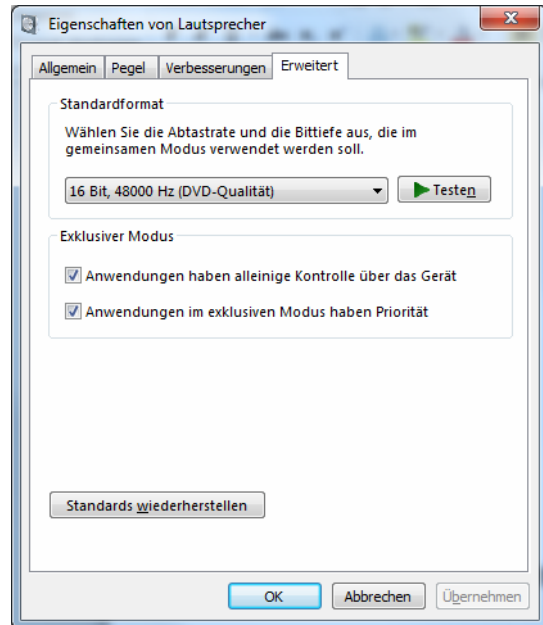
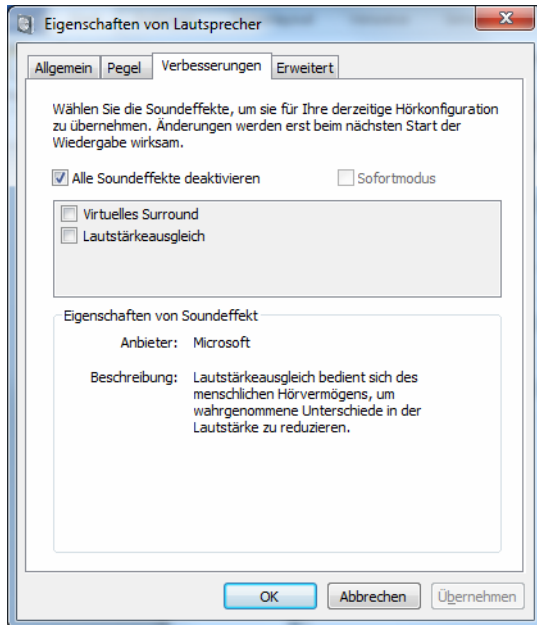


Wenn wir schon beim Überprüfen sind, dann vergleichen Sie doch gleich auch noch die übrigen Einstellungsseiten:

SS 2013

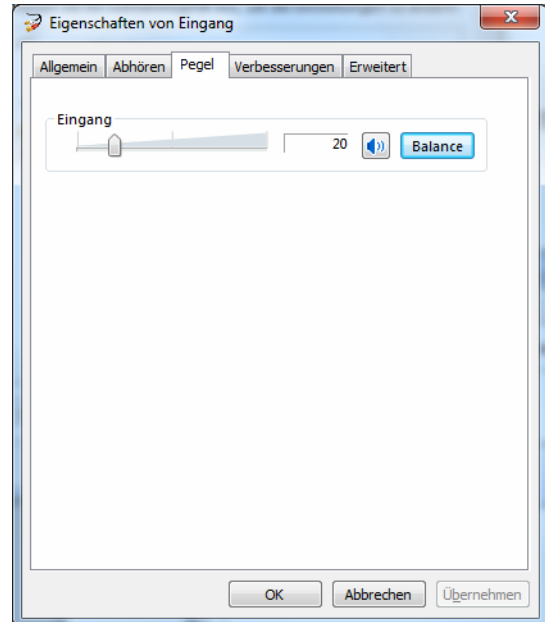
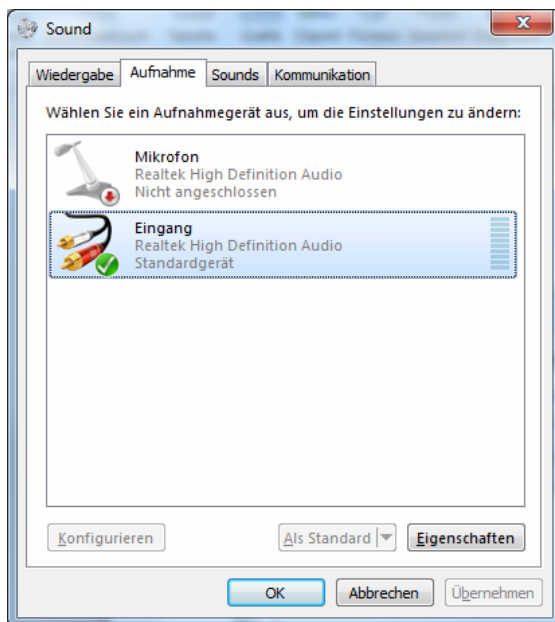
Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker



Die Einstellung der Eingangsempfindlichkeit erreichen Sie über die Seite Aufnahme und dort über den Button Eigenschaften des (Line-)Eingangs.

Wie oben bereits erwähnt und im Fenster unten zu sehen, muss hier i.d.R. eine niedrige Einstellung gewählt werden (ca. 15 % bis 20 %), um bei der späteren Messung einen Amplitudengang mit einem Durchlassbereich um 0 dB zu erhalten.

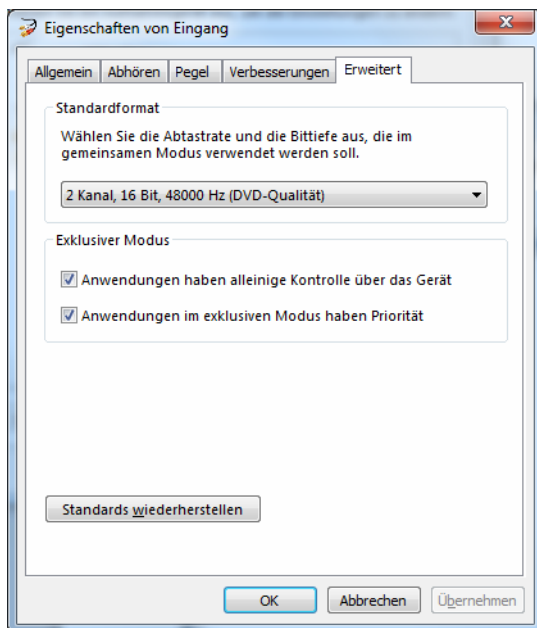
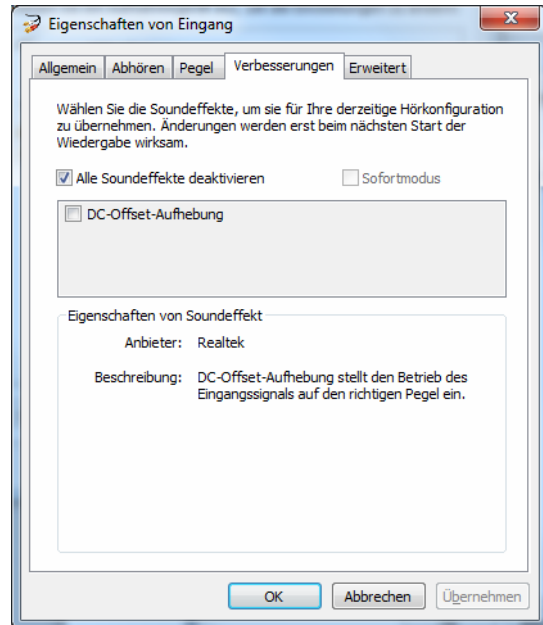
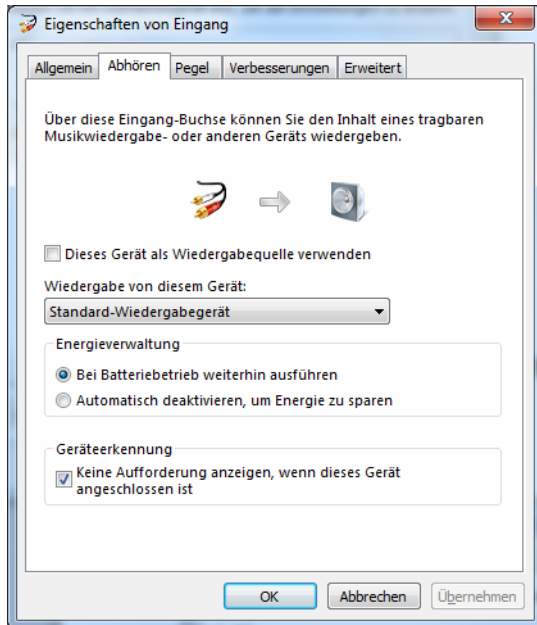


Auch hier sollten Sie noch die folgenden Einstellungsseiten überprüfen:

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker



Überprüfen Sie jetzt noch im Messprogramm audioTester die im unteren Bereich des Fensters für Output (Line-Out) und Input (Line-In) angegebenen Sampleraten und Bitbreiten. Hier sollten jeweils 48000 Hz und min. 16 Bit oder der Datentyp float gewählt sein. Falls nicht, können Sie über das Menü Options → Audio-In/Out-Parameter oder durch Doppelklick auf das jeweilige Feld die Einstellung abändern.

Über den Button „Setup“ lassen sich die Parameter für die Amplitudengangmessung einstellen. Starten Sie die Messung nach sinnvoller Parametereinstellung (z.B. für den Frequenzbereich) über den Button „Start“.

Stimmt der ermittelte Amplitudengang exakt mit dem von Ihnen erwarteten Verlauf überein? Erklären Sie mögliche Abweichungen.

SS 2013

Fakultät für Technik, Studiengänge EIT/TI

Prof. Dr.-Ing. Thorsten Benkner, Dipl.-Ing.(FH) Felix Becker

3.5 Durchführung:

Falls noch nicht geschehen, erweitern Sie die Filterfunktionalität jetzt auf Stereobetrieb (ersetzen/ergänzen des lokalen Filterergebnisses `sum` durch/um die Berechnung von `l_sum` und `r_sum` und die entsprechende Übergabe an die Transmit-Interrupt-Routine `data_word = (l_sum << 16) | (r_sum & 0x0000FFFF);`).

Geben Sie anschließend wieder eine Musik- oder Sprach-Datei über das DSK wieder und vergleichen Sie den Höreindruck mit und ohne Filterfunktion.

Ändern Sie dazu Ihr Programm entsprechend ab, so dass es wieder auf eine reine "Durchreicher"-Funktion wie in Aufgabenteil 1 reduziert wird. Anschließend bauen Sie das Filter wieder ein. Am direktesten erfolgt der Vergleich, wenn Sie eine **Umschaltmöglichkeit** zwischen Durchreichen und Filtern über z.B. den `USER_SW0` vorsehen (siehe Infoblatt „DSP Starter Kit DSK6713“), in dem Sie beispielsweise Ihr Filterergebnis überschreiben:

```
if (!USER_SW0) data_word = data_buffer[j];    // durchreichen
```

Testen Sie dabei auch die korrekte Stereofunktionalität (Links-/Rechts-Trennung) mit Hilfe der Balance-Schieber, die Sie über die oben abgebildeten Fenster erreichen.

Die Fragen sind handschriftlich in der Versuchsvorbereitung bzw. Ihrem Versuchsprotokoll zu beantworten!