

Praktikum Mikroprozessortechnik I

Versuch: ADSP-BF533 EZ-KIT Lite™

Thema 1: Einführung in die Arbeit mit dem Evaluierungssystem –
Analyse des Beispielprogramms Talkthrough_I2S

Versuchsziel:

In der Zielstellung des Versuchs sollen sich die Teilnehmer folgende Grundkenntnisse in der Arbeit mit dem DSP BF533 und dem Software-Entwicklungssystem VisualDSP++ aneignen:

- Projektverwaltung und die Programmentwicklung,
- die Anwendung von Debug-Funktionen,
- das Senden von Programmen an den Signalprozessor und
- der Start und die schrittweise Verfolgung ihrer Ausführung.

Diese Ziele werden durch das schrittweise Nachvollziehen des talkthrough – Beispiels erreicht. Gleichzeitig sollen sich die Teilnehmer dabei in Bezug zur elektronischen Schaltung des Evaluierungsboards auch die praktische Bedeutung der einzelnen Programmschritte, der Registerinhalte bei Konfigurationen, der Initialisierungen von Schnittstellen etc. erschließen. Die zugehörigen Informationen müssen in den Datenblättern des Prozessors, des Codec und des ADSP-BF533 EZ-KIT Lite aufgesucht werden.

0 Vorbemerkungen & allgemeine Informationen

Im praktischen Umgang mit dem ADSP-BF533 EZ-Kit Lite sind große Sorgfalt und Umsicht erforderlich. Der Nutzer ist bei Schäden, die durch Fahrlässigkeit oder auch infolge von mangelnder Vorbereitung im Umgang mit der Baugruppe entstehen, für deren Reparatur bzw. Ersatzbeschaffung (ca. 500 €) verantwortlich.

0.1 Der erste Schritt in der praktischen Versuchsausführung ist die **Prüfung der Voreinstellung** der DIP-Schalter Switch 9 für den I2S-Modus.

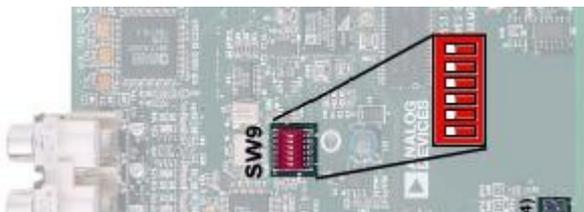


Abb. 1: Voreinstellung Schalterbank SW9

SW9-pin6 on,
SW9-pin5 on,
on = Schalterposition Richtung Audio-
Buchsenleiste der Platine, siehe Abb. 1.

Gegebenenfalls informieren Sie den Betreuer, die Umstellung der DIP-Schalter erfolgt durch ihn.

0.2 Was bedeuten diese Schalterstellungen ? - Allgemeine Informationen zum I²S – Modus:

Der serielle Datenausgang des A/D-Wandler ist auf das bekannte I²S Format voreingestellt, bei dem die Datenübertragung um 1 BCLK - Takt nach der LRCLK-Flanke verzögert beginnt. Ebenso gilt auch für den D/A-Wandler: „ serial data input mode defaults to I²S “. [AD1836]

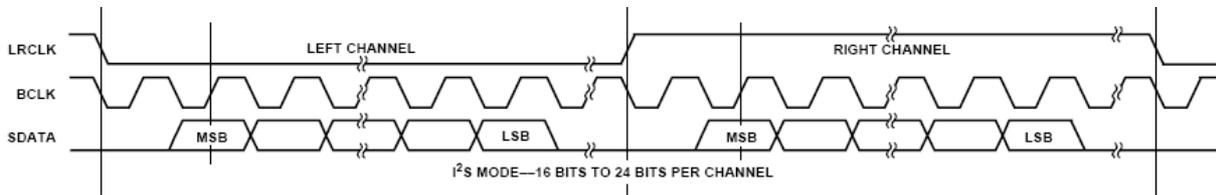


Abb. 2: Signaldiagramm für den I²S – Modus, aus [AD1836]

Im I²S – Modus gehören zu einem einzelnen Übertragungskanal mindestens 3 Leitungen, dies sind das Datensignal an sich, ein Datenbit – Clocksignal und ein Links/Rechts-Auswahlsignal. Pro Übertragungskanal kann damit in einer festgelegten Richtung (Receive oder Transmit) ein Stereo-Signal übertragen werden.

Zu einer seriellen Schnittstelle gehören immer TxD- und RxD- Leitungen, so daß pro Schnittstelle normalerweise ein Stereo - CODEC (Eingang & Ausgang, jeweils links & rechts) angeschlossen werden kann. Jede der beiden SPORTx-Schnittstellen des BF533 besitzt jedoch neben dem primären noch einen zusätzlichen sekundären Kanal. Daher können an einen SPORT 2 Einfach-Stereo – CODEC angeschlossen werden. Der AD1836 selbst besitzt 3 Stereo Aus- und 2 Stereo-Eingänge ($\approx 2 \frac{1}{2}$ „EinfachStereoCODEC“). Ein SPORT im I²S – Modus bedient davon also 2 Stereo Aus- und 2 Stereo-Eingänge, das dritte Stereo – Ausgangspaar bleibt folglich ungenutzt.

Eine alternativ mögliche Übertragungsart ist der Time Division Mode (TDM), bei dem alle Wandlerkanäle pro Richtung jeweils nur über eine einzige Datenbitleitung übertragen werden. Möglich wird dies durch die zeitlich aufeinanderfolgende Staffelung der Datenpakete.

0.3 Randbedingung bei Rechnerstart

Das Evaluierungsmodul darf noch nicht an die Betriebsspannung angeschlossen sein, wenn der PC gestartet wird. Erst nachdem Windows gestartet ist, kann der Kleinspannungsstecker angeschlossen werden.

Hinweis!

1 Vorbereitung

Das im Praktikum zu analysierende Programm *talkthrough* liest über den Codec AD1836A analoge Daten von den A/D – Wandlereingängen ein und gibt sie in unveränderter Form wieder über die D/A-Wandlerausgänge aus. Demzufolge werden zur Erzeugung und Messung der externen Signale mindestens ein Signalgenerator und ein Oszilloskop benötigt. Beide Geräte lassen sich auch mittels geeigneter Programme und der Soundkarte auf dem PC nachbilden.

Machen Sie sich zunächst mit der Hardwarestruktur des ADSP-BF533 EZ-Kit Lite vertraut. Nutzen Sie dazu die Beschreibung [ESM03] ADSP-BF533 EZ-KIT Lite Evaluation System Manual. Beantworten Sie sich dazu die nachfolgenden Fragen bzw. bereiten Sie sich auf die genannten Teilthemen vor.

1. Fertigen Sie sich eine Übersichtsskizze zur externen Beschaltung des Entwicklungssystems an. Wie und wo müssen Oszilloskop und Signalgenerator angeschlossen werden, damit jeweils über den linken Kanal der Wandleranschlüsse ADC1 Signale eingelesen und über DAC1 wieder ausgegeben werden können ?
2. Geben Sie die Art der Steckverbinder am Kabel an, wenn a) ein handelsüblicher Signal- oder Funktionsgenerator an das ADSP – Board oder b) ein über die PC-Soundkarte nachgebildetes Gerät angeschlossen werden soll. Notieren und begründen Sie, welche Einstellungen am Generator zu wählen sind, wenn der Kammerton A (440Hz) mit 50% der bei Vollaussteuerung des A/D-Wandlers möglichen Amplitude eingelesen werden soll. Die interne Verstärkerstufe sei dabei auf $V=1$ eingestellt.
3. a) Welche Belegung haben die drei Leitungen am Mikrophonanschluß der Soundkarte in Ihrem privaten PC? b) Welche Belegung haben die drei Leitungen jeweils am Line In - und Line Out - Anschluß der Soundkarte in Ihrem privaten PC? Antwort bitte mit jeweils Skizze und Spannungsangaben!
4. Untersuchen Sie durch praktische Test an Ihrem privaten Rechner die Möglichkeiten von Programmen, die mittels Soundkarte ein Oszilloskop oder einen Signalgenerator nachbilden. Realisieren Sie dabei u.a. die Einstellungen für die voranstehende Aufgabe.
5. Welche unterschiedlichen Parametrierungsmöglichkeiten bieten die analogen Ein- und Ausgänge des Codec und wie können die unterschiedlichen Parameter praktisch eingestellt werden? Beschreiben Sie Möglichkeiten zum experimentellen Nachweis der Wirkung dieser unterschiedlichen Einstellungen.
6. Wieviel Bits des A/D-Wandlers werden bei 50% -iger Aussteuerung effektiv genutzt ?
7. Welche Dateitypen gehören zu einem C – Projekt und was beinhalten sie? Beschreiben Sie die Schritte, mit denen aus C-Quellcode ein ablauffähiges C-Programm erzeugt wird?
8. Wie funktioniert die Interruptverarbeitung? Nennen Sie am Beispiel des PC die jeweils beteiligten Komponenten und beschreiben Sie detailliert den schrittweisen Ablauf. Verallgemeinern Sie dieses Verarbeitungsschema und informieren Sie sich in den Datenblättern über die Grundzüge der Interruptverarbeitung beim DSP.
9. Welche Interruptleitungen sind beim ADSP-BF533 EZ-KIT Lite an den CODEC AD1836A und welche an den DSP BF533 angeschlossen ?
10. Wozu dient ein DMA – Transfer?
11. Erstellen Sie nach Durchsicht des Datenblatts zum Flash – Speicher [PSD42] eine Skizze mit den wesentlichen Bestandteilen dieses Bausteins.
12. In den Schaltungsunterlagen zum ADSP-BF533 EZ-KIT Lite ist am AD1836A eine Reset-Leitung angeschlossen. Woher kommt diese Leitung? Stellen Sie eine Vermutung über die notwendigen Programmschritte zur Erzeugung des Reset - Zustandes am CODEC an.
13. Auf dem ADSP-BF533 EZ-KIT Lite befinden sich 6 LED. Untersuchen Sie die Schaltungsunterlagen mit dem Ziel, ihren Anschluß und damit auch Wege zur Programmierung der LED – Zustände herauszufinden.

2 Versuchsdurchführung und Aufgaben:

2.1 Arbeit mit VisualDSP++

Analysieren Sie zuerst die Programmschritte im Programm *talkthrough*, das hier mit einer Ansteuerung des Codecs AD1836A im I2S – Modus arbeitet. Bei dieser Aufgabe lernen Sie den Umgang mit häufig benötigten Werkzeugen von VisualDSP++.

Die eigentliche Funktion mit Ein- und Ausgabe von Signalen wird später in Abschnitt 2.2 untersucht.

Gehen Sie wie nachfolgend beschrieben zeilenweise durch das Programm. Erklären Sie sich dabei die Bedeutung und das Zustandekommen der einzelnen Initialisierungsschritte. Hierfür ist ein Vergleich der im Programm aufgeführten Registereinträge mit denen in den Dokumentationen, z.B. zum BF533 und zum AD1836A, vorgeschriebenen Werten sehr hilfreich und sinnvoll. Vermerken Sie für spätere Aufgaben die Positionen im Programmquelltext, an denen z.B. Routinen für eine digitale Filterung eingefügt werden müssen.

Gehen Sie nun wie folgt vor:

- Beim Start des PC muß das Evaluierungsboard noch spannungslos sein. Erst nachdem Windows gestartet wurde, kann auch das Board in Betrieb genommen werden. Der Treiber für die USB – Verbindung (nur für die Debug – Funktionen) wird automatisch geladen.
- Überprüfen Sie die korrekten Verbindungen zwischen ADSP-BF533 EZ-KIT Lite™ und PC. Beachten Sie, daß der Mikrofoneingang der Soundkarte nicht verwendet werden darf, da dann die Gefahr besteht, daß die Mikrofonversorgungsspannung auf einen ADSP-BF533 EZ-KIT Lite Ausgang eingekoppelt wird.
- Starten Sie die Software VisualDSP (in der jeweils auf dem PC aktuell verfügbaren Version), so wie dies in der Einführungsveranstaltung zum Praktikum demonstriert wurde.
- Starten Sie das Oszilloskop- und das Signalgeneratorprogramm. Die Version bzw. das Programm richten sich nach jeweils aktuellen Vorgaben oder der selbst getroffenen Auswahl. Prüfen Sie die regelrechten Verbindungen
 - (i) von (PC-) Oszilloskop (Anschluß Line_In der Soundkarte) und analogem Ausgang am ADSP-BF533 EZ-Kit Lite sowie
 - (ii) von (PC-) Funktionsgenerator und analogem Eingang.

Hinweis!

Die Einstellungen und Ergebnisse dieser beiden Programme werden erst im zweiten Teil der Versuchsdurchführung genauer betrachtet. Sie bleiben hier also zunächst unbeachtet.

- Legen Sie eine Kopie des in C realisierte Projekts *talkthrough_I2S* in Ihrem eigenen Nutzerverzeichnis an. Diese Projektkopie soll dann im Verlauf dieses einführenden Praktikums mit zusätzlichen Kommentaren im Quelltext zu einer Beispieldokumentation erweitert werden. In späteren Praktika nutzen Sie dieses eigene Ergebnis dann weiter.
- Starten Sie das Entwicklungssystem Visual DSP++.
- Öffnen Sie dann das vorher in das eigene Verzeichnis kopierte C - Projekt *talkthrough_I2S*

zur Codec – Anwendung, wie es in der Einführungsveranstaltung zum Praktikum demonstriert wurde. Es erscheint ein Schirmbild wie in Abb. 3 zu sehen.

- Verschaffen Sie sich zunächst einen ersten Überblick über die verfügbaren Funktionen und Steuerelemente der Software. Die wichtigsten Funktionen können auch direkt über Schaltflächen erreicht werden, die beim Anfahren mit dem Mauszeiger einen kurzen Hilfetext anzeigen. Lassen Sie sich alle(!) Hilfstexte anzeigen und notieren Sie sich davon die für Sie vermutlich wichtigen Funktionen, die Sie für das Übersetzen, das Senden an den DSP und den anschließenden Programmdurchlauf, der auch im Einzelschrittbetrieb kontrolliert werden soll, benötigen werden.
- Öffnen Sie danach gleichzeitig auch die Dokumentationsdateien (pdf – Dateien) a) zum EZ-KIT Lite [ESM03], b) zum BF533 [HWR533] und c) zum Codec1836A [AD1836] , um in diesen Datenblättern nach den Hinweisen, Erläuterungen und Begründungen für die programmtechnischen Details suchen zu können.

Nachfolgende Abbildungen zeigen Details und Bestandteile der Programmierumgebung:

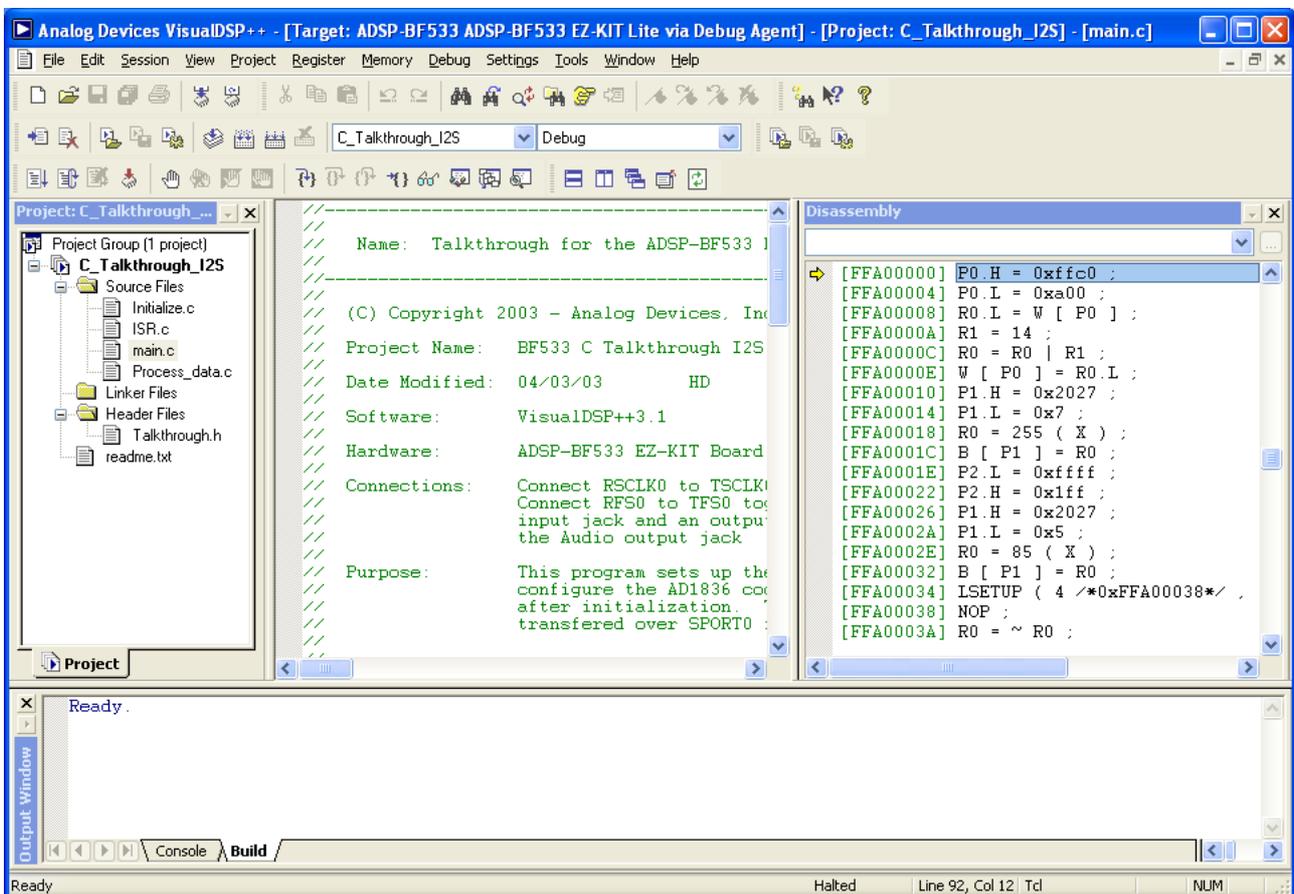


Abb. 3: VisualDSP++ unmittelbar nach dem Öffnen von C_Talkthrough_I2S.dpj

- Lassen Sie mit dem *Build File* – Befehl einzelne Dateien übersetzen und kontrollieren Sie die Ausschrift im unteren Meldungsfenster von VisualDSP++.
- Lassen Sie das gesamte Projekt mit *Build Project* übersetzen. Unmittelbar im Anschluß wird das ausführbare Programm an den DSP übertragen. Abb. 4 zeigt den resultierenden Programmzustand.

- Verwenden Sie die Befehle der Schaltflächenleiste mit den Debug – Funktionen und verfolgen Sie damit schrittweise die Programmabarbeitung. Erweitern Sie in den Quelltext dabei mit ausführlichen Kommentaren, Debug – Informationen und Details aus den Schaltkreis- und Systembeschreibungen.
- Fügen Sie bedarfsweise Hilfsvariablen ein, da mit den Debug - Funktionen nicht auf alle (vordefinierte) Systemvariablen zugegriffen werden kann. Lassen Sie diese Hilfsvariablen überwachen (mittels Schaltfläche Expression,) und überprüfen Sie die Werte durch Vergleich mit den in der Dokumentation [HWR533] aufgeführten Vorgaben. Im Anhang A und B sind die Adressen der in den Speicherbereich eingeschobenen Register (Memory Mapped Register, MMR) aufgeführt.
- Wiederholen Sie nach einem erneuten Übersetzen des Projekts o.g. Schritte. Nutzen Sie nun die von Ihnen eingesetzten Hilfsvariablen als zu überwachende Ausdrücke.

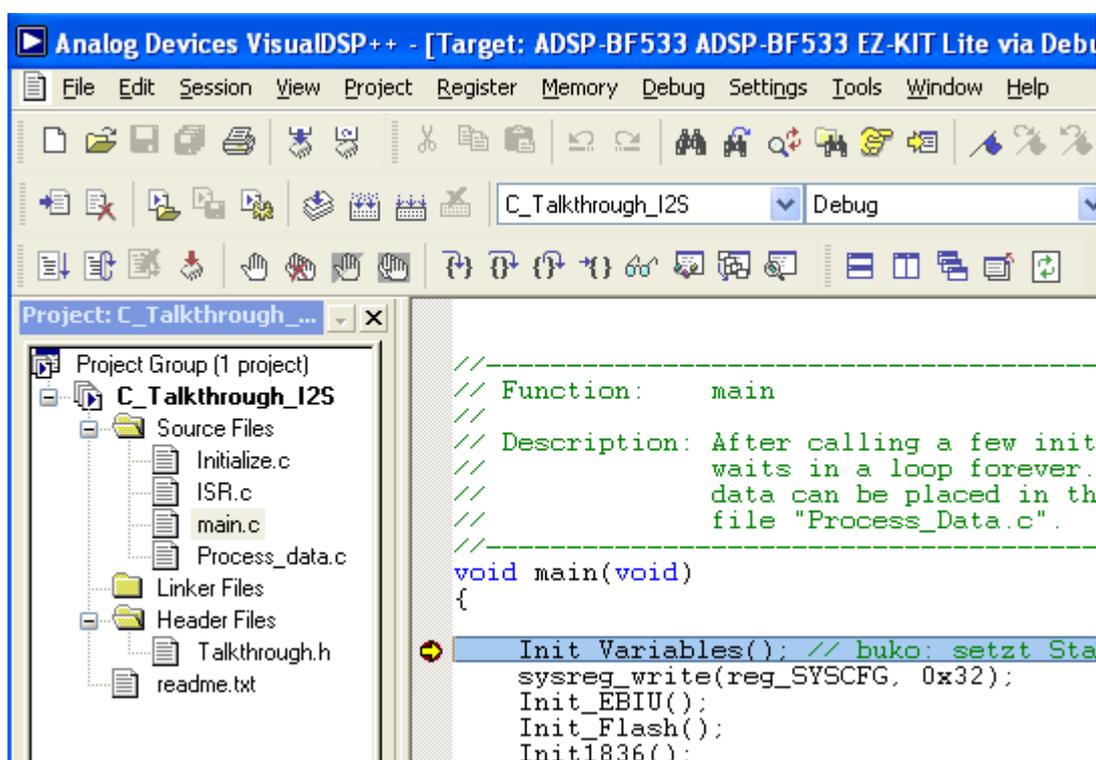


Abb. 4: Das neu übersetzte ausführbare Programm wurde an den DSP übertragen und stoppt an einem automatisch gesetzten Haltepunkt unmittelbar hinter dem Einsprung in die main - Funktion.

- Finden Sie heraus und notieren Sie sich, wie der Inhalt einzelner Speicherplätze überwacht und geändert werden kann.
- Machen Sie sich mit der Funktionsweise aller Schaltflächen aus Abb. 5 vertraut. Worin bestehen die Unterschiede der Funktionen?



Abb. 5: Schaltflächen zur vom Programmierer gesteuerten schrittweisen Programmabarbeitung und -überwachung.

- Betätigen Sie abschließend die Programmschaltfläche *Reset*.

2.2 Analyse der Programmfunktion

Die Bedienung der VisualDSP++ -Funktionen wurde in Abschnitt 2.1 erlernt.

Jetzt sollen die Ein- und Ausgabefunktionen im Programm *C_talkthrough_I2S* unter Beachtung des extern angeschlossenen Signalgenerators und des Oszilloskops untersucht werden. Machen Sie sich nun mit der Funktionsweise der zyklisch ablaufenden Einlese- und Ausgabevorgänge (hier noch ohne zwischenzeitliche Signalverarbeitung) vertraut.

Gehen Sie dabei wie folgt vor:

- Stellen Sie sinnvolle Werte in der Signalgenerator- und Oszilloskopsoftware ein. Ein- und Ausgänge wurden bereits in Pkt. 2.1 verbunden und geprüft.
- Laden Sie das Programm auf den DSP. Die Abarbeitung stoppt nach dem Einsprung in die Funktion *main()*, wie in Abb. 4 gezeigt. Starten Sie jetzt den kontinuierlichen Programmablauf mit der Schaltfläche *Run (F5)*.
- Am PC-Oszilloskop muß ein den Parametern des Eingangssignals entsprechendes Bild zu sehen sein. Falls nicht, finden und beseitigen Sie zunächst die Ursachen. Die weiteren Schritte setzen die regelrechte Funktion von Generator-, *talkthrough_I2S*- und Oszilloskopprogramm voraus.
- Betätigen Sie *Halt* und dann die *Reset* Schaltfläche. Das DAC – Ausgangssignal, dargestellt auf dem Oszilloskop, verschwindet.

- Laden Sie das Programm erneut auf den DSP. Die Programmabarbeitung stoppt in der Funktion *main()*. Gehen Sie nun in der Funktion *main()* mit *Step Over* schrittweise bis zur Programmzeile `while(1);` und beobachten Sie dabei nach jedem Schritt das Oszilloskopbild.
- Beobachtung: Auch nach mehrmaligem Betätigen von *Step Over* erscheint kein Ausgangssignal.
- Brechen Sie diesen Versuch ab. Betätigen Sie *Halt* und dann die *Reset* Schaltfläche.

Wo erfolgt eigentlich das Einlesen und die Ausgabe der Daten?

- Laden Sie das Programm erneut auf den DSP. Es wird wieder automatisch und erwartungsgemäß am Haltepunkt in der Funktion *main()* gestoppt.
- Betätigen Sie nun die *Run* Schaltfläche. Am Oszilloskop erscheint das vom D/A-Wandler ausgegebene Signal.
- Wechseln Sie beim laufenden Programmbetrieb durch einen Doppelklick auf den Dateieintrag *ISR.c* im linken Arbeitsbereichsfenster in den Quellcode der Funktion `EX_INTERRUPT_HANDLER(Sport0_RX_ISR)`
{

```
// confirm interrupt handling
*pDMA1_IRQ_STATUS = 0x0001;
...
```

Um welche Funktion handelt es sich hier vermutlich ? Haben Sie diesen Quelltext beim schrittweisen Programmdurchlauf bereits durchlaufen ?

- Betätigen Sie die *Halt* Schaltfläche.
- Wechseln Sie wieder in die Darstellung des Quelltextes von *ISR.c* . Setzen Sie dort in die Quelltextzeile

```
iChannel1RightIn = iRxBuffer1[INTERNAL_ADC_R1];
```

mit Doppelklick auf den linken grauen Rand einen Haltepunkt, der als roter Punkt erscheint.

- Betätigen Sie nun wieder die Schaltfläche *Run* . Die Programmabarbeitung stoppt am neuen Haltepunkt.

Warum wird diese Quelltextposition jetzt erreicht, nicht aber beim früheren schrittweisen Durchlauf ?

- Verfolgen Sie mit *Step Into* den weiteren Programmablauf. Notieren Sie sich die nun erreichten Programmpositionen und vergleichen Sie diese mit denen aus dem früheren Schrittbetrieb.

Überlegen Sie sich, an welcher Programmposition z.B. ein digitales Filter zur Signalverarbeitung eingefügt werden muß. Formulieren Sie schriftliche Lösungsvorschläge für die eigene Dokumentation dieser Praktikumsdurchführung bzw. der Versuchsaufgaben.

Nutzen Sie Ihre Beobachtungen zur Anfertigung einer grafischen Übersicht zum Ablauf der Programmabarbeitung im *C_Talkthrough_I2S* – Programm.