

Laborübung 3

Die Bearbeitung der Aufgaben erfolgt in Gruppen zu je max. zwei Teilnehmern. Alle Aufgaben (inkl. Skizzen, Rechnungen, Entwurfsschritte) sind schriftlich zu lösen und in einem Protokoll festzuhalten. Für den Hardware-Entwurf ist die Durchführung einer aussagekräftigen Simulation des Designs (oder Teilen davon) vor der eigentlichen Konfiguration des FPGAs obligatorisch, auch wenn dies in der Aufgabenstellung nicht ausdrücklich erwähnt ist. Alle Ergebnisse der Bearbeitung sind dem Dozenten im Rahmen der Gruppenrücksprache zu präsentieren. Für die Gruppenrücksprache zur Abnahme von Laborleistungen sind max. 2 Versuche pro Termin möglich (zur Vermeidung von trial-and-error Verfahren). Erfolgreiche Abnahmen werden auf dem Aufgabenblatt schriftlich quittiert. Eine erfolgreiche Laborteilnahme ist Voraussetzung für die Teilnahme an der Klausur am Ende des Semesters. Bitte bedenken Sie, dass eine gute Vorbereitung zu den Labortermen unabdingbar ist.

Aufgabe 1

In diesem Labortermin sollen Sie sich mit dem 24-Bit Audio Codec des DE2-115 Boards beschäftigen und auf verschiedene Arten einen Sound Generator entwerfen.

Tonhöhen und -intervalle lassen sich über Frequenzen und Frequenzverhältnisse beschreiben. Melodische Töne repräsentieren immer bestimmte Stufen der zugehörigen Tonleiter und die Intervalle sind Stufenverhältnisse. Jede Oktave einer Tonleiter ist in 12 Intervalle (Halbtöne im Abstand von Faktor $2^{1/12} = 1,0594$) unterteilt. Damit lassen sich alle Noten errechnen, sobald nur eine Frequenz bekannt ist. Die Oktaven werden aufsteigend durch Nummern gekennzeichnet, so umfasst z.B. ein Klavier 88 Tasten von Note A1 bis C8.

Im vorliegenden Fall soll von der C Dur-Tonleiter (C6 bis C7) ausgegangen werden, da die Frequenzen der Töne in den Bereich von 1 kHz bis 2 kHz fallen. Mit den folgenden vier Tönen (Frequenzen) können Sie bereits einfache Melodien spielen: C6 (1046,5 Hz); E6 (1318,5 Hz); G6 (1568,0 Hz); C7 (2093,0 Hz);

1. Machen Sie sich mit dem auf dem DE2-115 Board befindlichen Audio Codec Wolfson WM8731 und dessen Konfiguration mit I2C vertraut. Die Audio-Sampledaten werden jeweils über serielle Leitungen (getrennt für ADC und DAC) übertragen, die I2C-Schnittstelle lässt das Lesen und Schreiben auf die internen Konfigurationsregister zu. Diese Register sind im Datenblatt ausführlich beschrieben. Entwerfen Sie ein funktionstüchtiges VHDL Modul zur Ansteuerung des Audio Codecs vom FPGA aus. Dieses soll die Sound-Ausgabe über den Line-Out Kanal ermöglichen. Das ZIP-Archiv zu diesem Labortermin stellt ein geeignetes Gerüst bereit.

2. Entwerfen Sie nun ein VHDL Design zur Generierung der oben genannten Frequenzen und Auswahl mittels Taster (Button). Gehen Sie der Einfachheit halber von einem Rechtecksignal aus und berechnen Sie vor der Implementierung die vom Audio Codec max. verarbeitbare Signalamplitude. Ordnen Sie KEY3 der Note C6, KEY2 der Note E6, KEY1 der Note G6 und KEY0 der Note C7 zu und lassen Sie jeweils eine dem Taster zugeordnete LED bei Betätigung leuchten. Priorisieren Sie die Tonausgabe so, dass nur ein Ton auf einmal erzeugt wird, egal wie viele Tasten gedrückt werden (C7 → höchste Priorität, C6 → niedrigste Priorität).
3. Zeichnen Sie ein Blockschaltbild (auf Entity- bzw. Component-Ebene) Ihres Designs, aus dem neben der eindeutigen Kennzeichnung von Ein- und Ausgängen der Zusammenhang aller beteiligten Module (Sound Generator, WM8731 Ansteuerung, evtl. Clock PLL, etc.) klar ersichtlich ist.
4. Führen Sie die Implementierung Ihres Designs und die anschließende Konfiguration des Altera Cyclone-IV EP4CE115-F29C7 FPGAs mit Ihrem Design durch und testen Sie die Funktionsfähigkeit. Achten Sie dabei insbesondere auf die korrekte Zuordnung beim Pin Planning. Zum Testen stehen im Labor kleine Lautsprecher oder Kopfhörer zur Verfügung.
5. Freiwillig: Ergänzen Sie Ihren Entwurf um die Möglichkeit, die an den Codec übermittelte Signalamplitude (Lautstärke) mehrstufig per Schalter (Switch) einstellbar zu machen.

Aufgabe 2

Das generierte Rechtecksignal für die Ausgabe an den Audio Codec soll nun durch ein Sinussignal ersetzt werden.

1. Machen Sie sich mit Direct Digital Synthesis (DDS) vertraut. Nach diesem Prinzip werden Abtastwerte eines analogen Signals in einer Lookup-Table gespeichert und zyklisch ausgelesen. Der Altera Cyclone-IV EP4CE115-F29C7 FPGA verfügt über 3.888 kBit eingebetteten Speicher, der zur Realisierung der Lookup-Table genutzt werden kann. Die Eingangswortbreite der Lookup-Table richtet sich nach der Anzahl der gewünschten Stützstellen. Diese wiederum richten sich nach dem gewünschten Frequenzumfang der Schaltung. Führen Sie alle nötigen Rechnungen durch und zeigen Sie, ob und wie die oben genannten Frequenzen realisiert werden können. Eine besondere Eigenschaft des Sinussignals ist dessen Symmetrien (horizontal und vertikal). Lassen Sie dies in Ihre Berechnungen mit einfließen.
2. Entwerfen Sie nun ein VHDL Design zur Nachbildung eines analogen Sinussignales mittels DDS. Dazu können Sie eine Case-Anweisung mit entsprechenden Zuweisungen verwenden oder über einen Zähler auf den Index eines Arrays zugreifen. Wählen Sie eine aufwandsoptimierte Lösung. Hier zählt nicht nur die

Funktionalität, sondern auch eine möglichst kompakte Implementierung. Einen möglichen Implementierungsansatz finden Sie im ZIP-Archiv zu diesem Labortermin.

3. Führen Sie die Implementierung Ihres Designs und die anschließende Konfiguration des FPGAs mit Ihrem Design durch und testen Sie die Funktionsfähigkeit.
4. Nach erfolgreicher Inbetriebnahme legen Sie eine Tabelle mit den Implementierungsparametern (Anzahl genutzter LUTs, FFs, RAM Blöcken, etc.) des funktionierenden Designs an. Sie finden diese Angaben im Synthese Report in Quartus II.
5. Freiwillig: DDS ist natürlich nicht auf die Erzeugung reiner Sinussignale beschränkt. Zeigen Sie die Generierung anderer Signale, z.B. Dreiecksignal, Sägezahn oder Rauschen.

Ab hier bitte keine Eintragungen vornehmen!

Aufgabe 1	Aufgabe 2