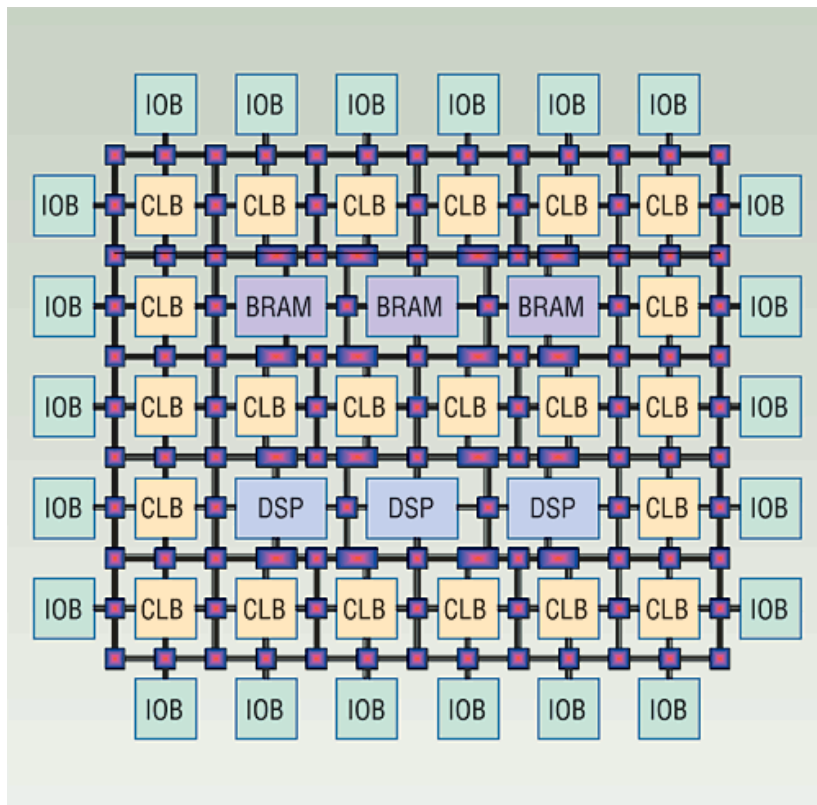


Laboraaufgaben - Teil 2



1 Lehrinhalt und Lernziele

Die vorliegende Laborübung(en) ergänzt die Vorlesung Programmierbare Logik. Empfohlene Voraussetzungen für diese Vorlesung und die dazugehörigen Laborübungen sind die Kenntnisse der Informatik, der Rechnerarchitektur, und der Digitaltechnik (und nicht zuletzt Ihr Interesse an der Thematik).

► Aufbau der Laborübung

Nach der Lehrveranstaltung sollten Sie in der Lage sein, einfache digitale Funktionen mit VHDL und/oder Verilog HDL beschreiben zu können und das Verhalten zu simulieren.

Die Laborübungen sind in mehreren Teilaufgaben unterteilt. Des weiteren wird vermittelt, wie einfache Ein- und Ausgabegeräte (hier Schalter und LEDS und Anzeigen) mit dem FPGA Chip verbunden werden.

Nr.	Inhalt	Punkte
L-1	Basiskomponenten	10
L-2	Takte, Zähler und Zeitgeber	10
L-3	Erkennung einer Bit-Sequenz	10
L-4	Booth-Algorithmus mit Datenpfad und Steuerwerk	10

► Laboraufgabe: Bearbeitung, Protokoll, Bewertung, Testat

Zum Ende der Vorlesung: Alle Laborübungen sind fehlerfrei zu bearbeiten und mittels Laborbericht (in Latex) zu dokumentieren. Es müssen alle 4 Laborübungen auf der Hardware in ihrer Funktion und Richtigkeit nachgewiesen werden.

Wenn Sie glauben alle Aufgaben erfolgreich abgeschlossen zu haben, wenden Sie sich an den Übungsleiter und bitten Sie um die Rücksprache. Der Übungsleiter wird Ihre Abgabe dann anhand der folgenden Faktoren bewerten:

- Korrekte Bearbeitung der Aufgaben
- Verständnis des eigenen Codes
- Codingstyle (ist der Code übersichtlich und der Stil kohärent?)
- Verständnis der generierten Netzliste
- Korrekte Beantwortung der Rückfragen des Übungsleiters

Ihre Aufgabe und der Laborbericht muss nach der Rücksprache als ZIP-Datei in das entsprechende Abgabefeld / Ordner in Moodle hochgeladen werden. Die erreichten Punkte werden Ihnen **erst nach upload** in Moodle eingetragen.

Hinweis: Zur Umsetzung der Aufgabenstellung lesen Sie bitte das *User Manual* https://www.terasic.com.tw/attachment/archive/502/DE2_115_User_manual.pdf durch.

2 Aufgabenstellungen L-2

In dieser Übung wird ein 3-Digit BCD-Zähler mit einer Ausgabe auf der 7-Segmentanzeige entwickelt. Dazu wird aus dem 50 *MHz* Takt des Intel DE2-115 Entwicklungsboard ein Kontrollsignal abgeleitet. Der Zähler soll im Sekundentakt seinen Zustand ändern. Der Taster KEY 0 wird als RESET für den 3-Digit BCD-Zähler verwendet.

Wenn Sie glauben alle Aufgaben erfolgreich abgeschlossen zu haben, wenden Sie sich an den Übungsleiter und bitten Sie um die Rücksprache. Der Übungsleiter wird Ihre Abgabe dann anhand der folgenden Faktoren bewerten:

- Korrekte bearbeitung der Aufgaben
- Verständnis des eigenen Codes
- Codingstyle (ist der Code übersichtlich und der Stil kohärent?)
- Verständnis der generierten Netzliste
- Korrekte Beantwortung der Rückfragen des Übungsleiters

Ihre Aufgabe und der Laborbericht muss nach der Rücksprache als ZIP-Datei in das entsprechende Abgabefeld in Moodle hochgeladen werden. Die erreichten Punkte werden Ihnen **erst nach upload** in Moodle eingetragen.

Hinweis: Legen Sie für jede Laborübung ein neues Quartus II Projekt an. Achten Sie darauf, dass der richtige FPGA-Typ als Zielbaustein ausgewählt ist (Cyclone IV EP4CE115F29C8).

Aufgabe 1 3-Digit BCD-Zähler

3 Pkt.

1. Legen Sie ein neues Quartus Projekt an. Schreiben Sie einen VHDL-Verhaltensmodell für einen 3-Digit BCD-Zähler. Überprüfen Sie den Zähler mit ModelSim.
2. Entwickeln Sie ein Strukturmodell. Fügen Sie diesen VHDL-Code ihrem Projekt hinzu.
3. Wählen Sie die richtigen Anschlüsse aus, um den 3-Digit BCD-Zähler auf drei Siebensegmentanzeigen anzuzeigen. Fügen Sie den VHDL-Code und die Anschlussbelegung dem Quartus Projekt hinzu (siehe auch Laborübung 1, Teil II).
4. Compilieren Sie den Quellcode und laden Sie die compilierte Schaltung auf das DE2-115 Entwicklungsboard. Verifizieren Sie ihren Entwurf.

Aufgabe 2 24h-Anzeige (Uhr)

4 Pkt.

1. Der obige Entwurf soll derart erweitert werden, dass eine **24h-Anzeige** der Uhr auf dem DE2-115 Entwicklungsboard zu sehen ist. Die Stunden (von 0 bis 24) werden dazu auf den Siebensegmentanzeigen HEX7-6 dargestellt, die Minuten (0 bis 60) auf der Siebensegmentanzeigen HEX3-2 und die Sekunden auf den (0 bis 60) auf den Anzeigen HEX3-2. Die Schalter SW15-0 werden zur Voreinstellung der angezeigten Uhrzeit verwendet.
2. Compilieren Sie den Quellcode und laden Sie die compilierte Schaltung auf das DE2-115 Entwicklungsboard. Verifizieren Sie ihren Entwurf.

Aufgabe 3 Stoppuhr

3 Pkt.

Der Entwurf aus dem vorherigen Aufgabenteil wird nun als Stoppuhr erweitert.

1. Legen Sie ein neues Quartus Projekt an. Schreiben Sie einen VHDL-Verhaltensmodell für die **Stoppuhr**. Überprüfen Sie die Stoppuhr mit ModelSim. Legen Sie selbstständig die Taster für die Start und Stoppfunktion fest.
2. Entwickeln Sie ein Strukturmodell. Fügen Sie diesen VHDL-Code ihrem Projekt hinzu.
3. Wählen Sie die richtigen Anschlüsse und BCD-Anzeige aus. Fügen Sie den VHDL-Code und die Anschlussbelegung dem Quartus II Projekt hinzu.
4. Compilieren Sie den Quellcode und laden Sie die compilierte Schaltung auf das DE2-115 Entwicklungsboard. Verifizieren Sie ihren Entwurf.

3 Hier ist Schluss!



Quelle: <https://www.flaticon.es/>