



Einführung in die Technische Informatik — VLSI-Systementwurf Praktikum: Stoppuhr

Martin Zabel
martin.zabel@tu-dresden.de
INF/1099

10. Oktober 2013

Aufgabe 1

Implementieren Sie in VHDL einen Dekoder für die Umwandlung einer 4-Bit-Binärzahl in die 7-Segment-Darstellung einer Hexadezimalziffer. Die Position der Segmente a bis g können Sie der Beschreibung des Praktikumsboards entnehmen. Beachten Sie, dass die Ansteuerung der Segmente low-aktiv erfolgt.

Für die Implementierung ist es zweckmäßig statt 8 Einzelsignalen (a bis g sowie Dezimalpunkt) einen 8-Bit-Signalvektor als Ausgangssignal vorzusehen. **Hinweis:** Nutzen Sie `case`- oder `select`-Statements zur Beschreibung des Dekoders.

Für die Eingabe der Binärzahl sind die Schiebeschalter SW3 bis SW0 zu nutzen. Steuern Sie nur das rechte Segment der 4-stelligen Anzeige an.

Hinweis: Die Aufgabe ist als Schaltnetz (ohne Taktsignal) zu lösen.

Überprüfen Sie die korrekte Funktion des Dekoders auf dem Praktikumsboard. Werten Sie die benötigten FPGA-Ressourcen im Praktikumsprotokoll aus.

Aufgabe 2

Implementieren Sie in VHDL ein Schaltnetz zur Bestimmung der Hamming-Distanz zweier 4-Bit-Worte.

Für die Ein- und Ausgabe sind zu verwenden:

| | |
|----------|-------------------------------------|
| Wort 1 | Schiebeschalter SW3 bis SW0 |
| Wort 2 | Schiebeschalter SW7 bis SW4 |
| Ergebnis | rechte Ziffer des 7-Segment-Blockes |

Überprüfen Sie die korrekte Funktion des Schaltnetzes auf dem Praktikumsboard. Werten Sie die benötigten FPGA-Ressourcen im Praktikumsprotokoll aus.

Aufgabe 3

Entwickeln Sie einen Modulo- n -Zähler, der bei einer Taktung mit 50 MHz einen Impuls pro Sekunde erzeugt. Die Impulslänge soll 1 Taktperiode betragen. Nutzen Sie die so erzeugte Impulsfolge zum periodischen Linksrotieren eines zyklischen 10-Bit-Schieberegisters um eine Bitstelle pro Sekunde.

Hinweise:

- Alle Register sind synchron mit ein- und demselben Taktsignal zu takten.
- Verwenden Sie einen additiven Operator zur Realisierung des Zählers.
- Bei einem Reset ist das Schieberegister mit einer ,1' an der rechten Stelle zu initialisieren.
- Realisieren Sie die Rotation durch eine geeignete Konkatination.

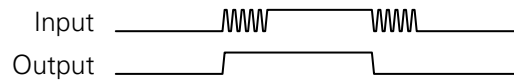
Für die Ein- und Ausgabe sind zu verwenden:

| | |
|-----------------|---------------------|
| Takt | 50 MHz |
| Reset | Schiebeschalter SW0 |
| Schieberegister | LED-Zeile |

- Bei welchem Zählerstand ist der Zähler zurückzusetzen? Wann muss die Ausgabe des Impulses erfolgen?
- Welcher Funktion entspricht der Impuls aus Sicht des Schieberegisters?
- Implementieren Sie das Schaltwerk in VHDL. Überprüfen Sie die korrekte Funktion des Impulses und des Schieberegisters im Simulator.
- Überprüfen Sie die korrekte Funktion des Schaltwerks auf dem Praktikumsboard. Werten Sie die benötigten FPGA-Ressourcen und die maximale Taktfrequenz im Praktikumsprotokoll aus.

Aufgabe 4

Die Taster des Praktikumsboards sind – abhängig von der Revision des Boards – nicht entprellt. Ihre Prelldauer beträgt bis zu 3 ms.



Waveform eines prellenden Eingangssignals und des entsprechenden entprellten Ausgangssignals.

Entwickeln Sie – unabhängig davon, ob ihr Board entprellte Taster besitzt – einen Automaten, der mit Hilfe eines Zählers nach einer am Eingang vom Taster erkannten Flanke für diese Dauer alle weiteren ignoriert und so ein entsprechend entprelltes Ausgangssignal liefert. Der Zähler ist hierbei mit dem Boardtakt von 50 MHz zu takten. Nutzen Sie diesen Entprellautomaten zum sicheren Ein- und Ausschalten einer LED jeweils durch den Druck desselben Tasters. Nutzen Sie für die Ansteuerung der LED einen zweiten Automaten.

Für die Ein- und Ausgabe sind zu verwenden:

| | |
|---------|-------------|
| Takt | 50 MHz |
| Eingabe | Taster BTN2 |
| Ausgabe | LED0 |

- Erstellen Sie jeweils State-Machine-Charts für den Entprellautomaten und den LED-Automaten. Abfrage und Steuerung des Zählers erfolgt durch den Entprellautomaten mittels geeigneter selbstdefinierter Signale.
- Welcher Typ von Automatenkopplung ist zu verwenden? Über welche Signale erfolgt die Kopplung?
- Implementieren Sie beide Automaten als getrennte VHDL-Module und überprüfen Sie die korrekte Funktion mittels Simulation.
- Implementieren Sie ein Top-Level-Modul welches beide Automaten miteinander koppelt. Überprüfen Sie die korrekte Funktion des Schaltwerks auf dem Praktikumsboard. Werten Sie die benötigten FPGA-Ressourcen und die maximale Taktfrequenz im Praktikumsprotokoll aus.

Aufgabe 5

Entwickeln Sie eine Multiplex-Ansteuerung für die 4-stellige 7-Segment-Anzeige des Erweiterungsboards um eine längere Zeichenkette auszugeben. Dabei soll jede halbe Sekunde der Text eine Stelle nach links verschoben werden.

Implementieren sie einen Zähler welcher Impulse mit geeigneter Frequenz ausgibt. Verwenden Sie für den Zähler den 50 MHz-Takt.

Zeigen Sie die Zeichenkette „HALLO“ an, wobei beginnend mit vollständig leerer Anzeige die Zeichenkette von rechts nach links die vier Stellen der Anzeige durchlaufen soll. Ist das letzte Zeichen nach links aus der Anzeige gewandert soll die Zeichenkette erneut ausgegeben werden.

Durch die Betätigung des Resets sollen alle vier 7-Segment-Blöcke gelöscht und danach wieder mit der Anzeige des ersten Zeichens der Zeichenkette in der rechten Stelle begonnen werden

Für die Ein- und Ausgabe sind zu verwenden:

| | |
|---------|---------------------|
| Takt | 50 MHz |
| Reset | Schiebeschalter SW0 |
| Ausgabe | 7-Segment-Block |

Gliedern sie ihren Systementwurf mindestens in drei Module: Top-Level, Decoder für Textzeichen und Multiplexer für den anzuzeigenden Teil der Zeichenkette.

- Wie viele Bits werden für die Kodierung eines Textzeichens benötigt? Kodieren Sie die Textzeichen mit einem eigenen Code und dokumentieren Sie diesen im Protokoll!
- Implementieren Sie Decoder und Multiplexer als getrennte VHDL-Module und überprüfen Sie die korrekte Funktion mittels Simulation.
- Implementieren Sie ein Top-Level-Modul welches beide Automaten miteinander koppelt. Überprüfen Sie die korrekte Funktion des Schaltwerks auf dem Praktikumsboard. Werten Sie die benötigten FPGA-Ressourcen und die maximale Taktfrequenz im Praktikumsprotokoll aus.

Aufgabe 6

Implementieren Sie eine auf Zehntelsekunden genaue Stoppuhr mit dem Boardtakt von 50 MHz als Referenz. Das Starten und Anhalten der Stoppuhr soll durch den Druck eines zu entprellenden Tasters ausgelöst werden. Das Rücksetzen der Stoppuhr soll durch das globale Reset erfolgen. Der aktuelle Stand der Stoppuhr ist dezimal auf dem 4-stelligen 7-Segment-Block auszugeben. Dabei sind eine Stelle für die Minutenzählung, zwei für die Sekunden und die verbleibende für die Zehntelsekunden vorzusehen. Verwenden Sie die Dezimalpunkte zur passenden optischen Unterteilung der Anzeige. Die Zeitmessung soll, entsprechend der verfügbaren Stellenzahl, „modulo 10 Minuten“ erfolgen.

Hinweise:

- Nutzen Sie einen BCD-Zähler pro Stelle mit einem entsprechenden Wertebereich.
- Kaskadieren Sie die BCD-Zähler mit Hilfe von Übertragsimpulsen.

Für die Ein- und Ausgabe sind zu verwenden:

| | |
|-------------|---------------------|
| Takt | 50 MHz |
| Reset | Schiebeschalter SW0 |
| Start/Stopp | Taster BTN2 |
| Ausgabe | 7-Segment-Anzeige |

- a) Überlegen Sie sich eine geeignete Zerlegung des Gesamtsystems in Teilkomponenten und spezifizieren Sie die deren Schnittstellen. Besprechen Sie kurz ihre Lösung mit dem Praktikumsbetreuer.
- b) Welcher Typ von Automatenkopplung ist zu verwenden? Über welche Signale erfolgt die Kopplung?
- c) Implementieren Sie die Teilkomponenten in VHDL. Erstellen Sie dazu für jeden Automaten ein State-Machine-Chart. Überprüfen Sie die korrekte Funktion jeder Teilkomponente mittels Simulation.
- d) Implementieren Sie ein Top-Level-Modul welches alle Komponenten miteinander verbindet. Überprüfen Sie die korrekte Funktion der Schaltung auf dem Praktikumsboard. Werten Sie die benötigten FPGA-Ressourcen und die maximale Taktfrequenz im Praktikumsprotokoll aus.

Zusatzaufgabe 7:

Erweitern Sie die Stoppuhr aus Aufgabe 6 so, dass über einen weiteren Taster eine Zwischenzeit genommen werden kann („Split“-Funktion).

Optional kann die Split-Taste zusätzlich zum Zurücksetzen des Zählerstandes (nachdem Zeitmessung gestoppt wurde) genutzt werden.

Für die Ein- und Ausgabe sind zu verwenden:

| | |
|-----------------------|---------------------|
| Takt | 50 MHz |
| Reset | Schiebeschalter SW0 |
| Start/Stopp (2. Zeit) | Taster BTN2 |
| Split (1. Zeit) | Taster BTN1 |
| Ausgabe | 7-Segment-Anzeige |

- a) Passen Sie die State-Machine-Charts und ihre Implementierung an. Überprüfen Sie die neue Funktion mittels Simulation.
- b) Überprüfen Sie die korrekte Funktion der Schaltung auf dem Praktikumsboard. Werten Sie die zusätzlich benötigten FPGA-Ressourcen und die maximale Taktfrequenz im Praktikumsprotokoll aus.