

# Praktikum zur Vorlesung „Hardwarebeschreibung“ „Digital-Design“

Prof. Dr.-Ing. habil. Jürgen Kampe \*

26. März 2025

## Praktikumsaufgabe

### Unsere Vorgehensweise

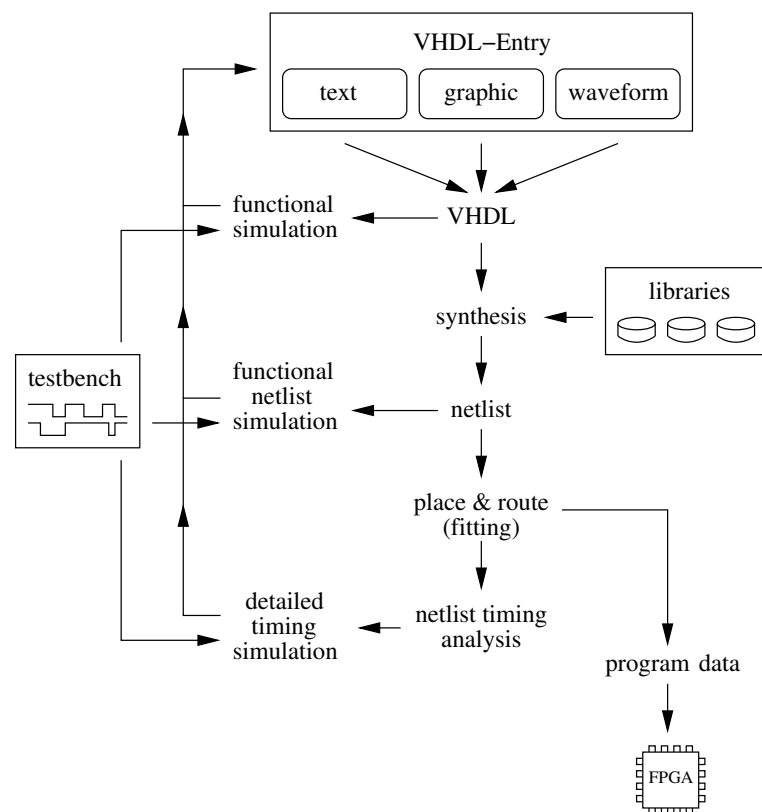


Abbildung 1: Entwurfsablauf in VHDL

### Spezifikation des Gesamtsystems

Die Aufgabe besteht darin, Verhalten und Schnittstellen des zu diesem Zeitpunkt nur informell spezifizierten Systems eindeutig in VHDL zu beschreiben, ohne zu diesem Zeitpunkt schon eine mögliche Realisierungsstruktur vorwegzunehmen.

\*Ernst-Abbe-Hochschule Jena, Institut für integrierte Systeme (IiS)

1. Zuerst werden die Datentypen und Konvertierungsfunktionen zwischen den Typen festgelegt, die zur Definition des Systems benötigt werden. Diese werden in einem Package definiert, falls nicht VHDL-Standardtypen oder Typen aus einem normierten Package (z. B. `std_logic_1164`) verwendet werden können.
2. Festlegung der Schnittstellen des Systems zu seiner Umgebung (Definition der top-level **entity**)
3. Beschreibung des Verhaltens in einer **architecture**. Um hier noch keine Aufteilung des Systems in Komponenten vorwegzunehmen, sollte diese nur eine Menge von Prozessen enthalten, aber keine Instanzen anderer Komponenten! In der abstrakten Verhaltensbeschreibung werden nur kausale Abhängigkeiten der Daten beschrieben (algorithmische Ebene).
4. Erstellen einer Testumgebung, mit der die Spezifikation simuliert werden kann. Ggf. müssen mehrere *use-cases* unterschieden werden!

### Funktionale Simulation

Mit Hilfe der Testbench erfolgt eine Verifikation der Spezifikation mit *ModelSim*.

### Entwurf mit VHDL

Es erfolgt die Implementierung des Systems:

- Partitionierung in Komponenten,
- Taktgenaue Modellierung des Systems und der Komponenten auf RT-Niveau.
- Erstellen einer Testbench zur Verifikation der Implementierung gegen die Spezifikation. Dazu muss das algorithmische Modell mit getakteten Registern gekapselt werden, so dass beide Modelle — die Spezifikation und die Implementierung — taktgenau simuliert werden können.

### Synthese

Das VHDL-Modell wird im Quartus-II kompiliert (Synthese + Place-&-Route + Timing-Analyse).

### Verifikation des Entwurfs mit einer funktionalen Simulation

Ohne Betrachtung des Zeitverhaltens wird die Implementierung funktional verifiziert. Es soll dazu die RT-Level Testbench verwendet werden.

### Simulation des Zeitverhaltens (*timing*)

Mit den Ergebnissen der *timing*-Analyse wird abschließend das Zeitverhalten überprüft. Wenn dazu Veränderungen an der Testbench erforderlich sind, dann muss diese gegen das RT-Modell verifiziert werden. Möglicherweise gibt es zusätzliche Spezifikationen des Zeitverhaltens, die in die Testbench aufgenommen werden müssen. Hierzu werden Konfigurationen verwendet.

## Vorstellung der Themenvorschläge

### Stoppuhr

**Aufgabenstellung** Es ist eine Stoppuhr mit folgenden Eigenschaften zu entwerfen:

- Ansteuerung einer 4-stelligen Siebensegmentanzeige,
- Messung von Zwischenzeiten,
- Pausenfunktion,
- Auflösung 1/10 Sekunde,
- Steuerung über drei Taster:

**Start-Stopp-Reset:** Mit dem ersten Tastendruck startet die Messung. Bei laufender Messung wird die Uhr angehalten und die Zeitdifferenz angezeigt. Der dritte Tastendruck bei angehaltener Stoppuhr setzt die Uhr zurück.

**Pause:** Per Tastendruck kann die laufende Zeitmessung unterbrochen bzw. fortgesetzt werden.

**Zwischenzeit:** Mit einem Tastendruck bei laufender Uhr wird die Zwischenzeit angezeigt so lange angezeigt, bis der Taster ein zweites Mal betätigt wird. In der Zwischenzeit läuft die Uhr weiter und kann auch bei angezeigter Zwischenzeit mit dem Start-Stopp-Reset-Taster angehalten werden.

Die Abbildung 2 zeigt die Ein- und Ausgänge der Stoppuhr.

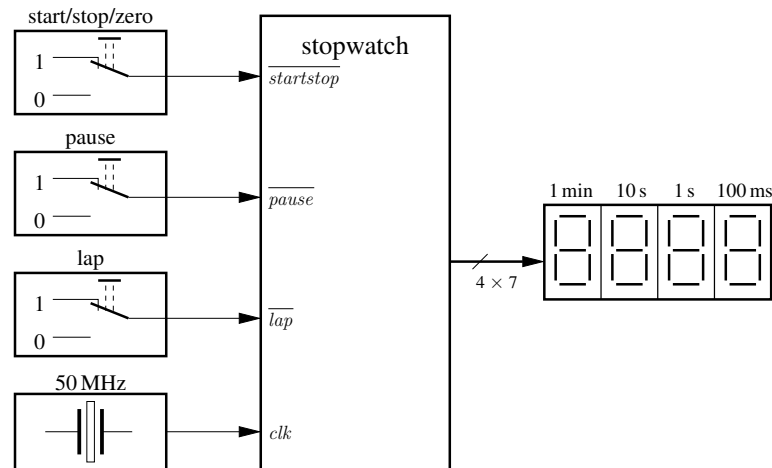


Abbildung 2: Schnittstellen der Stoppuhr

### Entwurf eines DDS-Signalgenerators (*Direct Digital Synthesis*)

**Aufgabenstellung** Es soll ein Sinussignalgenerator entworfen werden, der auf dem Prinzip der direkten digitalen Synthese beruht [1]. Das Gerät soll folgende Eigenschaften aufweisen:

- Referenzfrequenz:  $f_{\text{clk}} = 48,828 \text{ kHz}$
- Ausgangsamplitude mit 10 bit Auflösung
- Phasenauflösung: 12 bit
- Frequenzauflösung:  $f_{\text{res}} = 12 \text{ Hz}$
- Frequenzkontrollregister mit 10 bit Auflösung

Die Wahl der Ausgangsfrequenz erfolgt durch Eingabe eines Multiplikators mit Hilfe der Taster.

### Begriffsklärung

Frequenzauflösung: minimaler Abstand zweier vom Generator erzeugbarer Frequenzen  
 Phasenaufklärung: Feinheit der Zerlegung eines Vollwinkels

**Prinzip** Den allgemeinen Aufbau eines Signalgenerators basierend auf dem Prinzip der direkten digitalen Synthese zeigt Abbildung 3. Hauptbestandteile sind der Phasenakkumulator und eine Koeffiziententabelle in Form eines Speichers, in der die

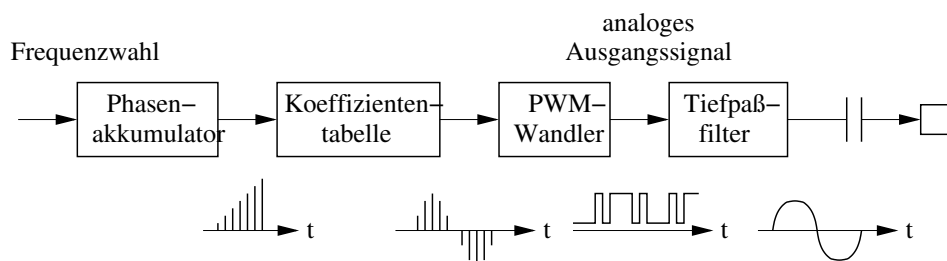


Abbildung 3: Blockschaltbild eines DDS-Signalgenerators

gewünschte Signalform abgelegt wird. Je nach gewählter Frequenz schaltet der rückgekoppelte Addierer die Phase schneller oder langsamer durch, dadurch entsteht die variable Ausgangsfrequenz. Am Ausgang der Koeffiziententabelle stehen die Amplitudenwerte der gewählten Kurvenform in digitaler Form zur Verfügung. Das Signal durchläuft anschließend die Blöcke DA-Wandler (10 bit Pulslängenmodulator PLM basierend auf einer Systemfrequenz  $f_{\text{sys}} = 50 \text{ MHz}$ ) und einem externen Tiefpaßfilter mit einer Grenzfrequenz  $f_g = 16 \text{ kHz}$ .

**Bestimmung der Generatorparameter** Die Frequenzauflösung des Synthesizers bestimmt sich durch die gewählte Phasenauflösung  $n$  und der eingespeisten Referenzfrequenz  $f_{\text{clk}}$  nach Formel (1).

$$f_{\text{res}} = \frac{f_{\text{clk}}}{2^n} \quad (1)$$

Die Phasenauflösung  $n$  wird auf 12 bit festgelegt. Mit der Referenzfrequenz von 48,828 kHz ergibt sich die geforderte Frequenzauflösung von 12 Hz.

Die Multiplikation von Glg. (1) mit dem im Frequenzregister eingestellten Faktor  $F$  bestimmt die resultierende Ausgangsfrequenz (2).  $F$  charakterisiert dabei die Schrittweite mit der der Adresszeiger inkrementiert wird.

$$f_{\text{out}} = F \frac{f_{\text{clk}}}{2^n} \quad (2)$$

Das Frequenzkontrollregister kann durch die 10 bit Ausführung Multiplikatorwerte im Bereich:  $0 \dots 1024$  annehmen. In der Einstellung 0 wird die Phase nicht inkrementiert, was zu einem Gleichsignal am Ausgang führt. Jedoch ist die Ausgangsamplitude von der, im Einstellmoment, anliegenden Phase abhängig, was in diesem Fall zu einem beliebigen Wert führt. Alle anderen Werte von  $F$  ermöglichen durch die vorgegebene Frequenzauflösung Ausgangsfrequenzen von 12 Hz bis 12,288 kHz in 12 Hz-Schritten.

## Literatur

- [1] Bar-Giora Goldberg, Digital Frequency Synthesis Demystified, LLH Technology Pub., Rom, 1999
- [2] Jouko Vankka, Methods Of Mapping From Phase To Sine Amplitude In Direct Digital Synthesis, Proceedings 50. Annual Frequency Control Symp., 1996, pp. 942-950.

## Messung der Reaktionszeit eines Probanden

**Aufgabenstellung** Es soll die Reaktionsgeschwindigkeit eines Probanden bestimmt werden. Dazu werden auf einem 7-Segment Anzeigepaar pseudozufällige Ausgaben erzeugt und die Reaktionszeit nach dem Erscheinen eines voreinstellbaren Anzeigemusters (auch möglich: nach Erscheinen einer Primzahl, nach Erscheinen einer Zahl der *Fibonacci*-Reihe oder ähnliches) bis zur Reaktion durch Drücken eines Tasters gemessen. Nach der Beendigung des Messzyklus durch Benutzereingabe werden die Anzahl der gültigen Versuche (d. h. die Reaktion erfolgte innerhalb des zulässigen Zeitbereiches nach dem Erscheinen des Anzeigemusters), die minimale, maximale und die durchschnittliche Reaktionszeit ausgegeben.

- Ansteuerung der Siebensegmentanzeige
- Steuerung über Taster
- Messbereich der Reaktionszeit 1 ms ... 1,999 s

## Anforderungen an den Praktikumsbeleg

Geben Sie Ihren Praktikumsbericht im Moodle termingerecht als zip-Archiv mit

- einem beschreibenden Textdokument (PDF) sowie
- mit allen VHDL-Dateien für den Entwurf einschließlich der Testbenches sowie
- der Clock-Constraints als *Synopsys Design Constraints* (\*.sdc) sowie
- dem PIN-Assignment für das FPGA-Board als *Quartus Settings File* (\*.qsf) sowie
- dem VHDL-Objekt für die *timing*-Simulation (\*.vho) und dem Zeitmodell im *Standard Delay Format* (\*.sdf) und
- der FPGA-Programmierung als *Quartus SRAM Object File* (\*.sof)

elektronisch ab. Zusätzlich soll die PDF-Datei mit: Namen, Matrikelnummer, Matrikel (z. B. 192E) und Thema eine von Ihnen unterschriebene Eigenständigkeitserklärung enthalten:

Hiermit bestätige ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken einschließlich Internetquellen entnommen sind, wurden unter Angabe der Quelle kenntlich gemacht.

Die Bewertung erfolgt an Hand der folgenden Kriterien:

1. Beschreibung des zu entwerfenden Systems in Text und Bild und Erläuterung des Lösungsansatzes,

2. Spezifikation der Testbench und Formulierung von *Use-Cases*,
3. kommentiertes und dokumentiertes VHDL-Modell und Testbench auf algorithmischer Ebene (Spezifikation des abstrakten Verhaltens),
4. VHDL-Testbench auf algorithmischer Ebene, ggf. für verschiedene *Use-Cases* (Dokumentation der Use-Cases),
5. funktionale Simulation mit *ModelSim*, Dokumentation des Ergebnisses,
6. kommentiertes VHDL-Modell auf RT-Ebene (Partitionierung, Implementierung der Komponenten),
7. erweiterte VHDL-Testbench für die Co-Verifikation der Implementierung gegen die Spezifikation,
8. funktionale Simulation der Implementierung, Dokumentation des Ergebnisses,
9. *timing*-Simulation der Implementierung mit Ergebnis.
10. Die Realisierung soll auf dem DE2-Evaluierungsboard funktionsfähig sein!