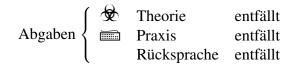
2. Blatt

## Fachgebiet Architektur eingebetteter Systeme **Rechnerorganisation Praktikum**



Ausgabe: 13. November 2017



## Aufgabe 1: NEQ4



Abbildung 1: Entity neq4

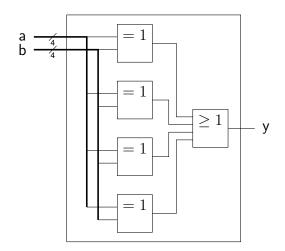


Abbildung 2: Architecture neq4

Name	Тур	in / out	Beschreibung
a	std_logic_vector(3 downto 0)	in	_
b	std_logic_vector(3 downto 0)	in	-
У	std_logic	out	Ergebnis des Vergleichs $a! = b$

In den folgenden Aufgaben sollen zu der oben gezeigten Antivalenzgatter-Schnittstelle mehrere Beschreibungsalternativen in Form von Architekturen auf *Logik*- und *Strukturebene* implementiert werden.

Dabei sollen alle Varianten des Antivalenzgatters identische Funktionalität aufweisen, d.h. nur für identische Eingangsvektoren a und b nimmt der Ausgang y einen *low* Pegel an, sonst ist dieser *high*.

- 1. Implementieren Sie zur Schnittstelle neq4 die Architektur logic in der vorgegebenen Datei neq4.vhd. Die Architektur soll den Ausgang y des Antivalenzgatters mittels der in VHDL vordefinierten *Logik* (Elementarfunktionen) aus den Komponenten der Eingangsvektoren a und b berechnen.
- 2. Implementieren Sie die Architektur netlist des Antivalenzgatters neq4 auf struktureller Ebene durch Instanziierung und Verdrahtung von geeigneten Elementargattern. Die Implementie-

- rung erfolgt ebenfalls in der Datei neq4. vhd. Greifen Sie dabei auf die geschriebenen zweistelligen Logikgatter aus dem 0. bzw. 1. Aufgabenblatt zurück.
- 3. Testen Sie ihre Implementierung mit Hilfe der vorgegebenen Testbench neq4\_tb. Rufen Sie dazu in dem Aufgabenordner der Vorgaben das Kommando make clean all in der Konsole auf. Die Testbench testet beide Implementierungen gleichzeitig. Sollten Sie den Test ausführen, während Sie nur eine Implementierung geschrieben haben, so können Sie die Fehler, welche die noch nicht geschriebene Implementierung betreffen, ignorieren. Die Datei neq4. vhd muss allerdings syntaktisch korrekt sein, damit die Simulation durchgeführt werden kann. Eine Betrachtung der Signalverläufe kann durch den Aufruf des Kommandos make view\_wave erfolgen. Im Betrachter finden Sie die Fehlermeldungen im unteren, rechten Fenster. Durch einen Klick auf den Fehler kann der Cursor im Signalfenster an die entsprechende Stelle verschoben werden.

## Aufgabe 2: 7-Segment-Treiber





Abbildung 3: Entity bin2Char

Abbildung 4: 7-Segment-Display

Name	Тур	in / out	Beschreibung
bin	std_logic_vector(3 downto 0)	in	binär kodierte Ziffer
bitmask	std_logic_vector(6 downto 0)	out	Steuerausgang für das 7-Segment Display.
			Eine '1' bringt die entsprechende LED im Dis-
			play zum Leuchten.

Binär-Kodierung	Hexadezimal-Zeichen
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	b
1100	C
1101	d
1110	Е
1111	F

In dieser Aufgabe soll ein bin2Char-Treiber für eine 7-Segment-Anzeige beschrieben werden, welche für jeden Eingangswert eine Bitmaske ausgibt. Diese Bitmaske wird zur Ansteuerung eines 7-Segment-Elements (vgl. 4) genutzt, sodass die hexadezimale Repräsentation des Eingangs auf dem Element angezeigt wird.

Für die Aufschlüsselung der Bitmaske (Bitmaskenposition zu LED-Position) betrachten Sie Abbildung 4.

- 1. Nutzen Sie die Datei bin 2Char. vhd aus den Vorgaben, um die Funktionalität zu implementieren. Legen Sie für das Mapping des Treibers von dem Eingang auf die Bitmaske ein array als constant an. Stellen Sie für die Implementierung zuerst eine Wertetabelle auf, in der Sie für jede Eingangskombination den Wert für die 7 Ausgangsbits festlegen. Diese Wertetabelle lässt sich dann direkt in das constant array übertragen.
- 2. Testen Sie Ihre Implementierung mit der vorgegebenen Testbench bin2Char\_tb durch die Benutzung des Makefiles. Rufen Sie dazu im Aufgabenverzeichnis der Vorgaben das Kommando make clean all auf.

## Literatur

- [1] Jonas Tröger Max Uffke Drechsler. Gtkwavemod-repository. https://gitlab.tubit.tu-berlin.de/jonas.e.troeger/GtkWave-Mod.
- [2] Mentor Graphics Corporation. *ModelSim SE Reference Manual*, 6.4a edition.
- [3] Mentor Graphics Corporation. *ModelSim SE Tutorial*, 6.4a edition.
- [4] Mentor Graphics Corporation. *ModelSim SE User's Manual*, 6.4a edition.