

Übung 6: Parametrisierung

Aufgabe: Kaskadenzähler

Ausgangspunkt ist ein Universalzähler, der mit 2 GENERICS parametrisiert werden kann:

- g_start: GENERIC-Typ: natural
- g_end: GENERIC-Typ: natural

PORTs des Zählers:

Name	Richtung	Typ	Bedeutung
clk	IN	std_ulogic	Systemclock, 1 Hz
reset	IN	std_ulogic	Reset (aktiv high)
enable	IN	std_ulogic	enable = '1': Zählen möglich enable = '0': Zähler bleibt stehen
carry	OUT	std_ulogic	Ist immer dann '1', wenn Zähler seinen Endwert erreicht hat
value	OUT	natural	Zählerwert

Der Zähler incrementiert "value" bei jedem Clock um eins, ausgehend von "g_start". Ist "g_end" erreicht, ist der nächste Wert wieder "g_start".

Aufgaben:

1. Implementieren Sie den Universalzähler mit den beiden GENERICS.
2. Simulieren Sie den Universalzähler mit Hilfe eines force-Files.
3. Der Universalzähler soll nun in einem konfigurierbaren Kaskadenzähler mit 2 Stufen eingesetzt werden, der eine Uhr mit Sekunden und Minuten implementiert. Definieren Sie dazu in einem PACKAGE "count_pkg" zwei Konstanten vom Typ ARRAY (0 TO 1) OF natural (oder allgemeiner ARRAY (natural RANGE <>) OF natural)
 - CONSTANT c_start_values: Start-Werte der beiden Stufen
 - CONSTANT c_end_values: End-Werte der beiden Stufen
4. Optional: Wählen Sie eine Beschreibung, die nur durch Ändern der Konstanten im PACKAGE aus der zweistufigen Uhr eine dreistufige (inkl. Stunden) machen kann.

ENTITY des Kaskadenzählers als Uhr:

Name	Richtung	Typ	Bedeutung
clk	IN	std_ulogic	Systemclock
reset	IN	std_ulogic	Reset (aktiv high)
values	OUT	ARRAY(natural)	Zählerwerte (Sekunden und Minuten resp. Sekunden, Minuten und Stunden)