

1. Aufgabe Erweiterung des 2-zu-1-Multiplexers

Erstellen Sie eine *unit* `TheNand2(Rt1)`, welche zwei Eingänge über ein *nand* mit einem Ausgang verbindet.

Erstellen Sie jetzt eine vierte *architecture* `StructTheNand2` des Multiplexers aus der vorigen Aufgabe. Diese soll ausschließlich *entity instantiations* von `work.TheNand2(Rt1)` enthalten. Ergänzen Sie die Struktur aus der vorigen Übung durch diesen vierten Multiplexer.

Untersuchen Sie die entstehende Schaltung mit dem *RTL-Viewer* und dem *Technology Viewer* (*Post Mapping*).

2. Aufgabe Multiplexer: Schneller beschreiben.

Obwohl der Multiplexer bereits auf vier verschiedene Arten beschrieben wurde, fehlen noch jene beiden Beschreibungsarten, welche in der Praxis die größte Rolle spielen.

Erstellen Sie eine fünfte *architecture* `UsingCase` des Multiplexers. Der Multiplexer soll mittels eines *case statements* beschrieben werden. Es sollen vorerst nur die Werte '0' und '1' berücksichtigt werden.

Beschreiben Sie den Multiplexer nun noch ein weiteres Mal mit dem *if-then-else statement* (*architecture UsingIf*).

Instanziieren Sie auch alle neuen Beschreibungen wieder aus Ihrer Prüfumgebung aus der vorigen Übung heraus.

Prüfen Sie die Funktion Ihrer Beschreibung mittels Simulation. Wie reagiert das Modell in der Simulation, wenn am Eingang `iSel` der Wert 'x' oder 'U' anliegt?

Ergänzen Sie die beiden neuen Designs so, dass für den Eingang `iSel` alle Werte in `std_logic` berücksichtigt werden. (Für die *architecture UsingIf* wird nun das *if-then-elsif-else statement* benötigt.)

Wie reagiert das Modell in der Simulation, wenn am Eingang `iSel` der Wert 'x' oder 'U' anliegt? Was ist Ihrer Ansicht nach die wünschenswerte Reaktion? Verbessern Sie Ihre Modellierung gegebenenfalls entsprechend.

Welche Schaltungen (*RTL-* im *Technology-Map Viewer*) erwarten Sie für die beiden Beschreibungen `UsingCase` und `UsingIf`? Wie sehen diese tatsächlich aus? Wie unterscheiden sich die beiden Versionen voneinander?

Testen Sie Ihren gesamten Entwurf auf dem *DE1-Board*. Suchen Sie sich hierzu passende LEDs für die Ausgänge aus.

3. Aufgabe Timinganalyse

In dieser Aufgabe geht es darum, welche Verzögerungszeiten die Beschreibungen der letzten Übungseinheit aufweisen.

- a) Zunächst bitte das Werkzeug TimeQuest Timing Analyzer starten (Uhrsymbol).
- b) Im TimeQuest-Unterfenster namens *Tasks* (unten links) im Ordner *Reports/Datasheet* den Report *Datasheet* anfordern (also mit Doppelklick starten).

Die Werte der Verzögerungszeiten (*Propagation Delay*) von den Ein- auf die Ausgänge werden in einer Tabelle angegeben. Die Spalten der Tabelle sind mit den Buchstaben R für *rising* und F für *falling* gekennzeichnet. Wie erklären Sie sich Bedeutung der vier unterschiedlichen Spalten? Warum fehlen einige Werte gänzlich? Woher stammt Ihrer Meinung nach der Name *Datasheet Report*?

Die Timinganalyse kann für unterschiedliche Situationen bei minimalen und maximalen Werten von Betriebsspannung (in mV) und Temperatur (in °C, abgekürzt mit C) durchgeführt werden. Die Auswahl wird im Unterfenster *Set Operating Conditions*, links oben vorgenommen. Vergleichen Sie die Werte und finden Sie die maximale und minimale Durchlaufzeit heraus.

Wechseln Sie mittels des Quartus-Menüeintrags *Assignments—Device...* auf ein geringfügig anderes FPGA und führen Sie die Timinganalyse erneut durch. Verwenden Sie ein FPGA, das mit C7 statt C6 endet. Um wie viel Prozent verändern sich die Werte nach erneutem *Compile* im Vergleich zum FPGA auf dem Board?

Probieren Sie erneut mit dem FPGA der Endung A7. Worin besteht der Unterschied in den Zeiten im Vergleich mit der Version C7?

Altera definiert vier Temperaturbereiche für seine Produkte. Werden diese eingehalten, garantiert Altera die Funktion unter Einhaltung der Ergebnisse aus der Timinganalyse.

C *Commercial*: 0°C to 85°C

I *Industrial*: -40°C to 100°C

A *Automotive*: -40°C to 125°C

M *Military*: -55°C to 125°C

Bei den Versionen ...C6, ...C7 und ...C8, ...A7, ...I7,... handelt es sich nicht um jeweils eigenen Produkten mit eigenem Herstellungsprozess. Wie könnte der Hersteller (Altera, TSMC) Chips nach dem Herstellungsprozess möglicherweise sortieren (und mit dem passenden Namen bedrucken und zum entsprechenden Preis verkaufen)?

Im Auswahldialog finden sich auch noch einige FPGA-Typen mit der Zusatzendung ES. Dies ist die Abkürzung von *Engineering Sample*. Es handelt sich dabei um Muster aus der Vorserie, die zur Entwicklung von Produkten vor der Verfügbarkeit von Serien-FPGAs gedacht sind.