**Dokumentation der Bauteilbibliothek für den Entwurf von Datenpfaden**

Bauteil **adder4** : 4-Bit Addierer, ohne Carry-In und Carry-Out

Bauteil **inverter4**: 4-Bit Inverter

Bauteil **const0000**: 4-Bit Konstante mit dem Ausgang “0000”

Bauteil **const0001**: 4-Bit-Konstante mit dem Ausgang “0001“

Bauteile **mem*A*x*B***: PROM-Speicher der Größe *A* Wörter mit jeweils *B* Bits. Erwartet Adresse am Bauteile **mem*A*x*B***: Eingang *Adr* und liefert die dazugehörige Speicherstelle am Ausgang  *Data*. Die Bauteile **mem*A*x*B***: Daten müssen in einem Textfile stehen, das im Arbeitsverzeichnis liegt. Pro Zeile Bauteile **mem*A*x*B***: in dem Textfile müssen die einzelnen Bits eines Speicherwortes ohne Leerzeichen Bauteile **mem*A*x*B***: zwischen den Bits stehen, ohne von Leerzeichen umschlossen zu werden. Die Bauteile **mem*A*x*B***: erste Zeile im Textfile wird an die Adresse 0 im PROM gelegt, die zweite Zeile an Bauteile **mem*A*x*B***: die Adresse 1 usw. Der Name der Textdatei kann im SchematicEditor von Xilinx Bauteile **mem*A*x*B***: eingegeben werden, indem das Attribut *FileName* mit dem Namen der Datei Bauteile **mem*A*x*B***: befüllt wird (Doppelklick auf das Bauteil im SchematicAdder öffnet einen Dialog Bauteile **mem*A*x*B***: mit den Attributen). Beispielzeile für ein mem16x8: 00010111

Bauteil **mux2x4**: 2-auf-1 Multiplexer, der jeweils 4-Bit breite Busse auf den Ausgang multiplext

Bauteil **mux4x4**: 4-auf-1 Multiplexer, der jeweils 4-Bit breite Busse auf den Ausgang multiplext

Bauteile **reg*X***: *X*-Bit breites Register, flankengesteuert (steigende Flanke auf Eingang *CLK*)

Bauteile **reg*X*WE**: *X*-Bit breites Register mit WriteEnable (‘1‘ auf *WE* erlaubt Schreiben, ‘0‘ verhindert Bauteile **reg*X*WE**:Schreiben), flankengesteuert (steigende Flanke auf Eingang *CLK*)

Bauteile **reg*X*WRWE**: *X*-Bit breites Register mit WriteEnable (‘1‘ auf *WE* erlaubt Schreiben, ‘0‘ Bauteile **reg*X*WEWR**: *v*erhindert Schreiben) und Reset (‘1‘ auf Eingang *RESET* beschreibt das Register Bauteile **reg*X*WEWR**:asynchron mit dem Wert “0000“, ‘0‘ bedeutet normaler getakteter Betrieb), Bauteile **reg*X*WEWR**:flankengesteuert (steigende Flanke auf Eingang *CLK*)

Bauteil **regAdd4**: 4-Bit breites Register mit inkrementierendem Eingang (Werte, die am 4-Bit breiten Bauteil **regAdd4**: Dateneingang *DataIn* liegen, werden um 1 erhöht, bevor sie in das Register Bauteil **regAdd4**: geschrieben werden) und WriteEnable (‘1‘ auf *WE* erlaubt Schreiben, ‘0‘ verhindert Bauteil **regAdd4**: Schreiben), sowie Reset (‘1‘ auf Eingang *RESET* beschreibt das Register asynchron Bauteil **regAdd4**: mit dem Wert “0000“, ‘0‘ bedeutet normaler getakteter Betrieb), flankengesteuert Bauteil **regAdd4**: (steigende Flanke auf Eingang *CLK*)

Bauteil **registerfile4x4RP1WP1**: Registerfile mit 4 Registern mit jeweils 4 Bit. Das Registerfile stellt Bauteil **register**einen Leseport (Registernummer über *RPAdr* einstellbar, Daten am Ausgang *DataOut* Bauteil **register**verfügbar), sowie einen Write-Port (Registernummer über *WRAdr* einstellbar, Bauteil **register***WE*=‘1‘ ermöglicht Schreiben, *WE*=‘0‘ verhindert Schreiben, Daten an *DataIn* Bauteil **register**anlegen) zur Verfügung. Register Nummer 0 enthält den Wert 0 und ist nicht Bauteil **register**beschreibbar, auch nicht bei gesetztem *WE*.