ELEC-E3540 Digital Microelectronics II – Exercise 3

Book: Peter J. Ashenden, "The designer's guide to VHDL", 3rd edition

Acquaint yourself with the following chapters: 4.1-4.4, 9.1-9.2, 14.1-14.2, 16.2.

Things to learn:

- Libraries

- Vectors and arrays

- for-if-generate

- Indexing

- Data I/O through CSV files

Pre-exercise tasks: Sketch the entity and architecture of a full-adder (1-bit adder). Input signals: A, B and CI (carry-in), type std_logic. Output signals: S and CO (carry-out), type std_logic.

Sketch the entity and architecture, in which an N-bit adder is implemented with a process. Input signals: addends A and B (N bits, type std_logic_vector), carry-in CI (1 bit, type std_logic). Output signals: sum S (N bits, type std_logic_vector), carry-out CO (1 bit, type std_logic).

Exercise task: Write a N-bit ripple-carry adder component (entity+architecture) without carry-in, built with N full-adders by using for-if-generate loop. Then, write a second N-bit adder component without carry-in, realized with a process and using "+" operator. Input signals: addends A and B (N bits, type std_logic_vector). Output signals: sum S (N bits, type std_logic_vector), carry-out CO (1 bit, type std_logic).

Write a testbench, which reads two 8-bit addends A and B from an input file at 10ns intervals. The input signals should be fed to two parallel designs under test DUT1 and DUT2, which are 8-bit adders implemented with the two different architectures described above. Output signals should be written to an output file.

Both input and output files should be formatted as comma-separated values (CSV) files. Each line of a CSV file consists of multiple fields, and the fields are separated by commas. For example, in our case, the input file might look like this:

```
 3,     17
90,    117
255,   0
...
```

Each line lists the values for A and B to be tested. The testbench should read one line at a time.

CSV files can be created and viewed with text editors, spreadsheet softwares, as well as exported/imported from Matlab.

Goal: Student learns the usage of libraries, vector types and their basic operations, signal arrays, for-if-generate loops, and data I/O through CSV files.

Workload: Preparations 4h + exercise 2h