ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG
INSTITUT FÜR INFORMATIK

Prof. Dr. Bernd Becker
Dipl. Inf. Tobias Schubert
M. Sc. (Eng.) Matthew Lewis

Mobiles Hardware-Praktikum
Summer Term 2003

Lab 4
Designing combinatorial and sequential circuits

Exercise 1

On the previous exercise sheet you used your 4-bit counter to make a "Running Light". Now, in this exercise you are going to use the 4-bit counter to output the values 1 to 6. This will allow you to simulate rolling a 6-sided die. The rolling process should be controlled with a push button input.

Implement this circuit in VHDL. Test and simulate the functionality of the circuit with respect to the EPM7128SLC84-15 FPGA.

Exercise 2

You should already have a good idea of what flip-flops are and how they are used. An individual flip-flop can store one bit of information and is the basis for larger memory organizations. Illustration 1 shows the schematic for an SRAM design that uses registers for memory cells (see *Hardware Design: Formaler Entwurf digitaler Schaltungen*, J. Keller und W. Paul, pg. 271). Using this type of memory structure and the MAX+PLUS II Baseline Software create an 8x2 bit SRAM. That is 8 separate memory addresses that can each hold 2 bits. The procedure for this lab is as follows:

1. First, use a memory organization like the one shown in Illustration 1 that contains 8 (2-bit) memory lines, 3 address lines, and 2 D-Latches in every cell.

2. Next design an address decoder that has a 3-bit input (address) and an 8-bit output. Remember to generate a symbol for this component. (Note: This is similar to a decoder you made in a previous lab)

3. Design the memory elements in VHDL. This component consist of 8 smaller 2-bit memory cells. Each memory cell has a Write input, Din[1..0] inputs, and Q[1..0] outputs. Simulate this design and then generate a symbol for it.

4. Now implement the output demultiplexer. This is the component (called On in Illustration 1) that takes the output from all the memory cells as an input and then outputs the selected 2-bit value. Again, produce a symbol of your circuit.

5. Once you have completed the design of your static 8x2-Bit SRAM, compile your circuit for the EPM7128SLC84-15 FPGA and simulate it.

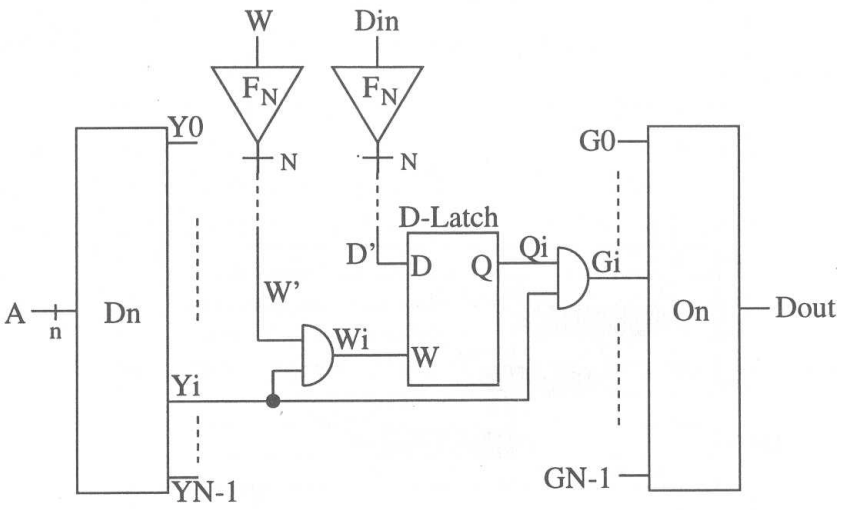

Illustration 1: Schematic structure of a static SRAMs

When your group is finished with the lab you must submit all GDF, SYM, VHD and SCF files. Please make sure the names of these files are appropriate (eg. myFullAdder.GDF). Also, please compress your final submission into one ZIP file before submitting it using the web page portal.