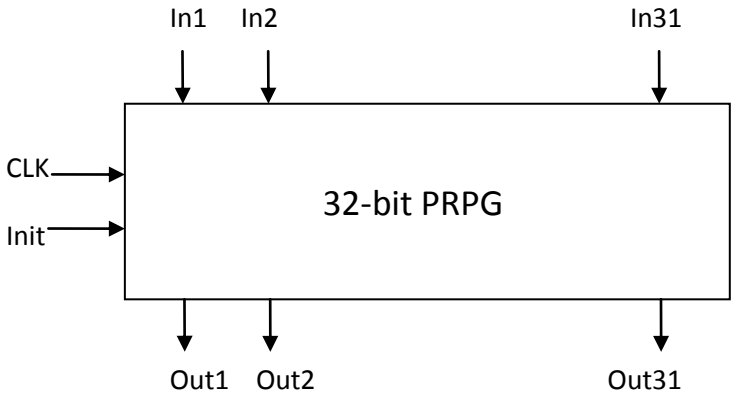
Project 2
ECE-470 Digital Design II

Project description

The objective of this project is to design and implement a 32-bit pseudo random pattern generator (PRPG) using VHDL (you can use Verilog instead if you prefer). The PRPG consists basically of a LFSR that generates pseudo-random vectors that could be used for BIST purposes for example. The block diagram of a PRPG is shown in the figure below:



The PRPG should have the capability of seed loading. This is done by holding ***Init*** high in the first clock cycle. When ***Init*** is high, the PRPG is loaded with values from the inputs. ***Init*** becomes and remains low from the second clock cycle on. Starting from the second clock cycle, the PRPG generates a random 32-bit vector every clock cycle. Use the following generating polynomial to design the structure of the circuit:

$$f(x) = x^{32} + x^{28} + x^{27} + x + 1$$

The code (VHDL or Verilog) for the PRPG should be simulated. You will need to submit your code written in the so called *structural description* style, demonstrate the simulation, and show waveforms for the initial random vector (seed) being loaded into the PRPG and its operation through 16 subsequent clock cycles.

Main steps that you should follow

- Download and install Altera's Quartus II Web Edition (v10.0). It is free. It is the same tool that is used in ECE-375. This is the preferred software because of the extra-credit options below. However, you are free to use any other CAD tool that has a VHDL/Verilog compiler and simulator. Quartus II Web Edition (v10.0) can be downloaded from here: ***https://www.altera.com/support/software/download/sof-download_center.html***
- Read the provided tutorial to learn how to use the tool (those who already know it can skip this).
- Write the VHDL code for your PRPG in order to implement its functionality as described in the section above.
- Create simulation stimuli for the first 16 clock cycles and simulate your design.
- Verify your PRPG on the DE2 boards in the lab. In this case, you should use the switches and LEDs available on the board to do seed loading and "display" of the least significant bits of the generated vectors.

Deliverables (no late submissions will be accepted)

(**Due on Dec. 6 2011, 100 points**):

- Your VHDL/Verilog code **50 points**)
- Demonstration of simulation and hardware implementation for the first 16 clock cycles (**50 points**)

Extra credit (optional)

Extra-credit b (3% of final grade for this course): Search and download from the internet the VHDL/Verilog code of a 16 bit adder. Then, write the VHDL/Verilog code for a testbench that should "host" your PRPG, the adder, and a memory array (storing the correct/expected outputs) such that all together will implement the BIST technique. The testing should be done only for 16 test-vectors (the memory will store only 16 expected output responses).