

## Bubble Sort - Power Estimation

### Motivation and introduction

In this exercise, you will learn how to estimate the power consumption on gate-level accuracy for different processors and different applications which were generated in the previous sessions. Then you will compare the power consumption for a certain clock period. We will use the basis CPU and the CPU which supports the “bgeu” instruction. The applications which will be used are basis bubble sort algorithm “bs\_basis.s”, the bubble sort algorithm which uses bgeu instruction “bs\_bgeu.s” and the optimized version of the bubble sort algorithm “bs\_bgeu\_opt????.s”. Before you start the exercises, read the Power Estimation tutorial to get a close view about the software tools that will be used in Chapter 7 of the Laboratory Script. The different processors and applications, mentioned previously can be combined to form 3 versions as following:

**Version1:** basis CPU “dlx\_basis.pdb” with basis bubble sort algorithm: “bs\_basis.s”

**Version2:** CPU which supports the instruction bgeu “dlx\_bgeu.pdb” with bubble sort algorithm which uses bgeu instruction: “bs\_bgeu.s”

**Version3:** CPU which supports the instruction bgeu “dlx\_bgeu.pdb” with optimized version of the bubble sort algorithm: “bs\_bgeu\_opt????.s”

For every part, that starts like “a)”, “b)” ... you have to mail the answers and asked files with a CC to your group members to **asip00@ira.uka.de** and use the topic “asipXX-Session2”, with XX replaced by your group number.

### Readings for the next session: Chapters 8

### Exercises

1. Create three different project directories in your “ASIPMeisterProjects” directory for three versions and set the “env\_settings” accordingly.
2. For all the above three versions, generate ASIPmeister hardware and software description files and simulate with ModelSim. During simulation also generate the VCD files for mentioned frequencies (first with 50MHz and then with Max. Frequency found in the last session).
3. Create Xilinx ISE project to estimate power with xPower for three versions as discussed in Chapter 7 of the Laboratory Script.
  - a) For all 3 versions determine the total and dynamic power.
  - b) Compute the total execution time (ms) for every version. You can use execution as the # of cycles multiplied by the clock cycle in ModelSim.
  - c) Compute the energy required for every version. Fill in all these results in the table below e.g. *PowerReport.xlsx* or *PowerReport ods*.
  - d) Does using “bgeu” instruction minimize the required energy? Version3 uses an application which needs less number of clock cycles than Version2; is it also power and/or energy optimized version compared to Version2?
  - e) Repeat a-d, but instead of taking the default of 50 MHz, use the individual maximum CPU frequency on which a CPU can run (You can get it from ISE\_Benchmark). This frequency has to be configured in tb\_ASIPmeister.vhd (search for CLK\_HALF\_PERIOD; e.g. 10 ns half period = 20 ns period = 50 MHz). XPower will automatically load this frequency from the VCD file.

Sample PowerReport.xlsx or PowerReport.ods

	Total Power [mW]	Dynamic Power [mW]	Execution Time [ms]	Energy [nJ]
<b>Version1</b> 50 MHz				
<b>Version2</b> 50 MHz				
<b>Version3</b> 50 MHz				
<b>Version1</b> Max. Freq: -----MHz				
<b>Version2</b> Max. Freq: -----MHz				
<b>Version3</b> Max. Freq: -----MHz				