ASIP Laboratory - Session 6

Paul Georg Wagner   Majd Mansour

June 19, 2017

Exercise 1: Exercise

In this session, the power and energy requirements of the various bubble sort solutions from the last sessions should be estimated. In order to do so, we first created three project directories and copied the necessary files from the last sessions. The first bubble sort version to be tested is the unoptimized implementation for the unmodified CPU. The second version uses the `bgeu` instruction of the modified CPU. Finally, the optimized bubble sort algorithm from session 4 is tested.

For all three versions we created the ASIPMeister projects and generated the VHDL files for the respective CPUs. Then we compiled the implementations with `make sim` and then used ModelSim to generate VCD (value change dump) files for each version. The VCD files contain the switching activities that occur when the respective project is executed on the FPGA board. Of course, the switching activity depends on the frequency, with which the board is operated. We first generated VCD files for a board frequency of 50 MHz and then for the respective maximum possible frequency.

Finally, we created a new Xilinx ISE project and generated a bit stream for each version. Afterwards we used the xPower tool to calculate the dynamic and leakage power requirements for the different versions.

a) *For all 3 versions determine the total and dynamic power.*

The xPower tool outputs the dynamic and the leakage power requirements for the analyzed program versions. The results are displayed in table 1.

Project	Frequency	Total Power	Leakage Power	Dynamic Power
dlx_basis	50 MHz	1109 mW	1043 mW	66 mW
dlx_bgeu	50 MHz	1109 mW	1043 mW	66 mW
dlx_bgeu_opt	50 MHz	1125 mW	1044 mW	81 mW

Table 1: Power requirements of bubble sort versions at 50 MHz.

The dynamic power requirement of a certain algorithm corresponds to the amount of work the processor performs in a certain amount of time. This explains why the dynamic power requirements of *dlx_basis* and *dlx_bgeu* are the same. The *dlx_bgeu* version uses the `bgeu` instruction instead of the `slte`/`bnez` instruction pair, which ultimately results in fewer instruction cycles, but does not change the amount of work that is done in a certain amount of time. The optimized *dlx_bgeu_opt* version, however, uses far less NOP instructions and hence maximizes the amount of work done in a certain time period. Hence the required dynamic power is higher than in the other version. On the other hand, the leakage power is the same for all versions. This is because the used FPGA board does not have power gating, so the leakage power for the whole board is always consumed. Since the leakage power is always consumed, the total power requirement of the optimized solution is only slightly higher than the total power requirement of the basis version.

b) *Compute the total execution time (ms) for every version.*

The execution times of the different program versions depend on the used FPGA frequency as well as the number of cycles that the program takes to execute. The *dlx_basis* version requires 6946 cycles to execute, which results in an execution time of $\frac{6946}{50\text{ MHz}} \approx 138,92\ \mu$s with a 50 MHz frequency setting. The *dlx_bgeu* reduces the amount of necessary instruction cycles to 4212, which results in an execution time of $\frac{4212}{50\text{ MHz}} \approx 84,24\ \mu$s. The optimized version even achieves a reduction down to 2492 cycles, which gives the execution time as $\frac{2492}{50\text{ MHz}} \approx 49,84\ \mu$s.

c) *Compute the energy required for every version.*

The amount of energy that is needed for execution depends on the power requirement and the execution time of the respective program. The energy can be calculated through $E = P * t$. The results are displayed in table 2.

Project	Frequency	Total Power	Execution time	Energy
dlx_basis	50 MHz	1109 mW	138,92 μs	154,06 μJ
dlx_bgeu	50 MHz	1109 mW	84,24 μs	93,42 μJ
dlx_bgeu_opt	50 MHz	1125 mW	49,84 μs	56,07 μJ

Table 2: Energy requirements of bubble sort versions at 50 MHz.

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?*

When using the `bgeu` instruction instead of the `slte`/`bnez` instructions, the amount of energy that is required for execution decreases from 154,06 μJ to 93,42 μJ. This is because the use of the `bgeu` instruction decreases the execution time while it does not change the power requirements compared to the *dlx_basis* version. With only 56,07 μJ the optimized version *dlx_bgeu_opt* requires an even less amount of energy to execute. Even though the optimized version requires more power – elicited by a higher switching activity – the total amount of energy for the execution is far lower, since the required execution time also decreased significantly. Hence the third version is not at all optimized for power, but instead for execution time and energy.

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.*

After the calculations for the different programs with a FPGA frequency of 50 MHz have been completed, the analysis should be repeated for the respective maximum possible frequency. In order to get the maximum possible frequency for each program, the Xilinx ISE tool can be used to calculate the critical paths and the resulting timings. This has already been done in the last session. The maximum possible frequency for the *dlx_basis* version yielded to 158 MHz, while the maximum frequency for the *dlx_bgeu* and *dlx_bgeu_opt* version is 142 MHz. These values result in execution times of $\frac{6946}{158\text{ MHz}} \approx 43,96\ \mu$s for *dlx_basis*, $\frac{4212}{142\text{ MHz}} \approx 29,66\ \mu$s for *dlx_bgeu* and $\frac{2492}{142\text{ MHz}} \approx 17,55\ \mu$s for *dlx_bgeu_opt*.

In order to retrieve the power and energy requirements of the programs when running with maximum speed, we first used ModelSim to create new VCD files for the respective programs. Since the VCD files contain the switching activity for the programs, they have to be updated for the increased FPGA frequency. This can be done by changing the `CLK_HALF_PERIOD` value in the ModelSim project file. Afterwards we executed the xPower tool with the updated VCD files and calculated the respective energy requirements. The results are shown in tables 3 and 4.

Project	Frequency	Total Power	Leakage Power	Dynamic Power
dlx_basis	158 MHz	1222 mW	1045 mW	177 mW
dlx_bgeu	142 MHz	1182 mW	1044 mW	138 mW
dlx_bgeu_opt	142 MHz	1223 mW	1045 mW	178 mW

Table 3: Power requirements of bubble sort versions at maximum frequency.

Project	Frequency	Total Power	Execution time	Energy
dlx_basis	158 MHz	1222 mW	43,96 μs	53,72 μJ
dlx_bgeu	142 MHz	1182 mW	29,66 μs	35,06 μJ
dlx_bgeu_opt	142 MHz	1223 mW	17,55 μs	21,46 μJ

Table 4: Energy requirements of bubble sort versions at maximum frequency.

Similarly to the results at 50 MHz, the leakage power requirements are the same for all versions. Also, the dynamic power requirement for the *dlx_bgeu_opt* version is again higher than the *dlx_bgeu* version. However, in contrast to the previous test, the dynamic power requirement of the *dlx_basis* version just as high as for the optimized version (177 mW). This is due to the higher frequency of 158 MHz that has been used for the *dlx_basis* version instead of the 142 MHz for the optimized version.

Just like in the previous tests, the *dlx_basis* version again requires the most energy to execute, even though the execution time has been reduced disproportionately because of the higher frequency. When comparing *dlx_bgeu* and *dlx_bgeu_opt*, the optimized version again requires far less energy, even though it has the greater power requirement, because the execution time is significantly reduced.

In summary it is apparent that the introduction of the `bgeu` instruction does not change the power consumption compared to the basis version of the CPU, while simultaneously reducing the number of instruction cycles, the execution time and the required energy. The optimized program version results in a higher switching activity on the FPGA board, and hence has a higher dynamic power requirement. However, since the number of cycles and the execution time is significantly reduced, the total amount of required energy for the algorithm is also reduced. Finally, running the FPGA board at the maximum possible frequency is beneficial for the energy requirement, since the execution time is reduced more significantly than the dynamic power requirement rises.