

## **Recommended Outline of the ECE 699 Project Specification Spring 2016**

### **1. Names of Team Members**

You are allowed to work on the project individually or in a group of two students. Group projects are expected to be more substantial and require approximately twice as many man-hours. The same final project score will be awarded to both teammates. Only individual projects can be used to fulfill the scholarly paper requirements.

### **2. Proposed Title**

### **3. Introduction and Motivation.**

Placement of the problem in the broader research area. Why is this project worth working on? Why is it original? Why is it useful? Why is it of interest to you?

### **4. Implemented Algorithm**

Please provide a list of references (one may be sufficient) describing the implemented algorithm using a pseudocode, diagrams, and/or informal description. Please note that you do not need to copy any text or diagrams from these references.

### **5. Previous Work & State-of-the-Art**

Please perform a comprehensive literature search looking for papers, books, presentations, web sites, and other resources describing:

- a. Necessary theoretical background
- b. Software/hardware implementations of the same or similar algorithm in Xilinx Zynq, Altera SoC FPGAs, Microsemi SmartFusion2 SoC, and on any other similar or older-generation platform
- c. Hardware implementations of the same or similar algorithm in FPGAs or ASICs
- d. Software implementations of the same or similar algorithm, with the focus on embedded system implementations.

### **6. Software Implementation**

C implementation is highly recommended. Please specify whether you are planning to implement the selected algorithm in C by yourself, or you are planning to use any existing implementation. In case you are planning to use an existing implementation, please provide the exact reference, including the list of authors, their affiliations, name of the program, language, library dependencies, license type, etc. In case, the program is written in a language different than C, you may decide to translate it to C by yourself.

### **7. Hardware Implementation**

VHDL implementation is highly recommended. Verilog implementation is permitted. The hardware implementation does not have to cover the entire algorithm. Your software/hardware implementation may offload only the most time-critical portion of

the algorithm to hardware. Please specify whether you are planning to implement the selected algorithm in RTL VHDL or RTL Verilog by yourself, or you are planning to use any existing implementation. In case you are planning to use an existing implementation, please provide the exact reference, including the list of authors, their affiliations, name of the design, language, library dependencies, license type, etc.

#### **8. HLS-ready C/C++/System C Implementation**

Please specify whether you are planning to develop the HLS-ready C/C++/System C implementation of the selected algorithm by yourself, or you are planning to use any existing implementation. In case you are planning to use an existing implementation, please provide the exact reference, including the list of authors, their affiliations, name of the program, language, library dependencies, license type, etc.

#### **9. Software/Hardware Partitioning**

Please provide your initial thoughts on the possible hardware/software partitioning schemes. Please note that it is highly recommended that you specify, implement, and test at least two different hardware/software partitioning schemes.

#### **10. Target Platform**

Please describe whether you are planning to use ZYBO or ZedBoard for your implementation and testing.

#### **11. Performance Metrics**

Please specify what performance metrics you are planning to use to characterize your implementations. Examples include the end-to-end execution time, the end-to-end throughput, resource utilization (in CLB slices, LUTs, registers, DSP units, BRAMs), etc. It is highly recommended that as a part of your project you determine ratios of performance metrics for RTL-based design methodology vs. HLS-based design methodology.

#### **12. Verification Method**

Please describe all methods you are planning to use to debug and verify the correctness of your implementation, including, but not limited to: generation of test vectors (please specify how?), simulation, run-time printouts, using input-output devices (buttons, LEDs, VGA port, HDMI port, etc.), using Integrated Logic Analyzer, etc.

#### **13. Possible Extensions (optional)**

- Running your application under Linux
- Using domain-specific languages and tools, such as Matlab/Simulink
- Others (please specify)

#### **14. Plan of Experiments**

Please specify a tentative plan of your experiments, including, but not limited to:

- a. Profiling
- b. Measurements of the end-to-end throughput and/or latency

- c. Measurement of the communication overhead between ARM Cortex A9 and your custom hardware accelerator
- d. Measurements of the maximum possible clock frequency in hardware
- e. Measuring dependence of the end-to-end throughput and/or latency on the location of data: OCM (On-Chip Memory) vs. external DDR3 SDRAM (Double Data Rate 3 Synchronous Dynamic Random-Access Memory)
- f. Speed of the bare-metal vs. Linux-based implementation (optional).

### **15. Tentative Schedule**

The proposed schedule, including intermediate goals to be achieved by the dates of the oral progress reports: March. 16-17, March 30-31, April 13-14, April 27-28.

### **16. References**

A comprehensive list of references, formatted using the IEEE Citation Style Guide, available at <http://libguides.nps.edu/citation/ieee>