PROJECT PROPOSAL
CSE 8351 Computer Arithmetic
PROPOSAL DUE: April 2, 2018
FINAL PROJECT DUE: May 14, 2018 before 3:00PM

A class project proposal written form for your class project in IEEE conference paper format, two-column, 10pt Times single-spaced with one-inch margins on all sides (MS Word and LaTeX templates are available from the IEEE website). Class projects are to be accomplished on an ***INDIVIDUAL*** basis. Your proposal and final project report *will be graded on style and grammar* as well as technical content (30% on style and content). The best way to make sure your proposal and report conform to the proper style is to read as many published conference papers as possible.

Each project will involve the implementation of an arithmetic circuit and/or algorithm. You may implement your circuit using HDL/Synthesis, Spice netlist or layout, X86 or ARM assembler, or as a C program. Note, if you choose the C program option, I expect low-level code using bit-level operators such as &, |, <<, etc. You will not be allowed to use libraries or code obtained from the web or other sources – all source code should be your own. I will use methods to search for authenticity of your source code and I will give ZERO credit if I determine that you used source code that was not your own personal creation.

If you choose a hardware implementation, then I expect you to evaluate your algorithms with a simulator and to report on cost (area) and delay metrics. If you choose to implement your algorithms in assembler or C, I expect to see functional simulations to validate correctness and an analysis of asymptotic worst case spatial (memory) and temporal (delay) complexities.

I expect to see either an advanced arithmetic circuit/algorithm that perhaps combines some of the basic methods we have discussed in class, or an advanced method that was not covered in class. If you decide to combine some of the methods in class, then you should provide motivation as to why you chose those methods. I will not allow more than one project idea per person. So if two or more people choose to propose the same acceptable project, I will let the person who wrote the best proposal have the project and the others must choose a different project.

Your ***two to three*** page proposal should have the following content:

1.0 INTRODUCTION

This section will describe the circuits/algorithms you are going to design for your project. It will contain any background information to explain the problem to an engineer WHO MAY NOT be familiar with computer arithmetic topics. It will provide the important references to past work done in the area (at least five are required). It will contain a description of WHY this circuit is important and how the design would be useful.

2.0 IMPLEMENTATION

This section will describe how you intend to implement your circuit. It should discuss the architecture (eg. a hybrid adder using CLAs and carry-skip adders) and contain block diagrams. For a hardware implementation, you should also describe your CAD tool flow, how you plan to simulate your circuit and which test vectors will be used. Finally, you should discuss the technology that will be used for realizing your circuit. (eg. static CMOS, domino logic, ECL, etc.). For a software implementation, you should describe your testbench source code and the test cases to be used for simulation, including how corner cases were chosen, why they were chosen, and why you feel they provide sufficient coverage.

3.0 SCHEDULE

This brief section will describe your schedule for accomplishing your project. Remember, projects will be due on May 14, 2018 at 3PM. You should describe your major milestones and when they are expected to be complete.

4.0 REFERENCES

This section will include the references to the major works in the past. You should have at least a five of these. This will mean using the library or ordering legitimate engineering and scientific papers from the web. I expect to see references to articles in IEEE Transactions journals (computer) and/or to conference proceedings for the major conferences in this field (ICCD, ARITH, etc), not just textbook references, references to class notes, my papers only, or non-refereed webpages. The IEEE Xplore feature should prove to be helpful for finding references.

The proposal is due to me by Monday, April 2, 2018 at 2PM. You will email the proposal to me as a single .pdf file and the time stamp of the email will serve as proof of when you submitted your report.

When the project is due, you will need to provide a written report (pdf) in the style of an IEEE conference paper (5-10 two-column 10pt Times Roman). Any source code developed should be included as a separate appendix and it should accompany your .pdf report as a .txt file that can be compiled or synthesized. You will email these files to me as a single zipfile or tarball and the time stamp of the email will serve as proof of when you submitted your report.

Part of the Proposal assignment is for you to demonstrate that you are capable of finding relevant papers in past work on your topic. You should have some recently published work in your reference list.

PROJECT PROPOSAL PROCESS

Each person must work individually on a project. All class members will work on different individual projects. You should choose a project from the list below. If two or more people propose to work on the same project, I will evaluate the proposals and allow the person with the best proposal to do that project. The others will have to propose a different project. I will accept project proposals that are not on this list, but you must get my approval of the proposal before you can continue on with the project.

SUGGESTED PROJECT TOPICS

1. Design and implement a 32-bit Wallace tree multiplier that uses signed binary adders internally.
2. Design and implement a circuit that computes a transcendental function such as sin, cos, log, etc. based on the CORDIC algorithm.
3. Design and implement a 32-bit array multiplier that uses radix-4 Booth encoding.
4. Design and implement a circuit that computes the square root of a 32-bit value.
5. Design and implement a 32-bit circuit that uses a ROM reciprocal table and a Newton-Raphson iteration to implement a transcendental function.
6. Design and implement a 32-bit SRT divider circuit.
7. Design and implement a floating point add/subtract unit using single precision IEEE 754 standards including support for ALL traps.
8. Design and implement a 32-bit selectable adder such that control lines allow you to select between a CLA, carry-skip, conditional-sum or prefix adder. Compare the designs in terms of area and delay.
9. Design and implement a MAC (multiply-accumulate unit) that uses RNS internally.
10. Design and implement a RNS division circuit that uses a CRT lookup-table.

11. Design and implement 3 different 32-bit integer addition circuits and experiment with pipelining to find the best possible throughput for each.

12. Design and implement a circuit that converts from polar to rectangular form and vice versa for complex values where the word size is 16-bit signed and fixed point values with eight integer and eight fractional bits.

13. Design and implement a circuit that computes the eigenvalues of a 5 by 5 matrix where each matrix entry is a 16-bit signed and fixed point value with eight integer and eight fractional bits.