EE244
October 8, 2013
Project Proposal

Title: Logic Energy And Resiliency (LEARN) tool

Group Members: Stevo Bailey and Amudhan Venkatesan

Problem Definition: Given a target FIT[1] rate, what is the minimum amount of energy a processor can consume?

Challenge: Energy conservation has arisen as the dominant focus of circuit and processor design, especially as designs become bigger and more complex. Resiliency is a metric of growing concern as chips scale, and is important for specific applications, like space-based circuits. Prior research investigates the tradeoffs between energy and resiliency from different levels of hierarchy [1]. Our project will investigate scaling these FIT rate and energy tradeoff calculations to large microprocessor and circuit designs.

Impact: We seek to limit overdesign and consume only enough energy as is needed to achieve the target FIT rate. We will produce a scalable design metric model that allows quick FIT and energy tradeoffs to be evaluated for any size processor or circuit.

Approach:

1) Explore the possible parameters that trade energy with soft error resiliency. For instance, by leaving some registers unprotected, energy is saved but resiliency is worsened. Other parameter ideas include leaving combinational logic unprotected, exploring different ways of mitigating soft errors (e.g. BISER vs. Razor vs. DICE vs. rad hard devices, etc.), voltage scaling, and performance scaling.

2) Evaluate the effect of selected parameters on a given processor's energy. Commercial CAD tools can estimate the energy of the processor both with and without the soft error resiliency techniques. These tools already scale well with large designs. The processor will be the general purpose Rocket core with the RISC V ISA developed at Berkeley. The technology may be the 28nm ST UTB FDSOI, or some other available PDK.

3) Evaluate the effect of selected parameter on a given processor's FIT rate. FIT rate will (hopefully) be estimated using existing flows developed at Berkeley. These will need adjustments to calculate the FIT for each parameter selected. They will also require adjustments to scale well for large designs; these scaling algorithms represent the primary work and contribution of our project.

4) Compile the results into a Pareto optimal curve. Draw conclusions regarding the effect of each parameter on energy, and how to optimize a design given a target FIT.

Timeline and Labor Division:

Week	Task	What do we need?	Who?
Oct. 14-18	Research and decide parameters to vary	Papers	Stevo and Amudhan
Oct. 21-25	Obtain and set up existing flows for FIT rate calculation Set up CAD tool flow to work with Berkeley processor (and smaller models first) and technology	Existing flows from Berkeley, CAD tool licenses? Synopsys licenses, can use existing CS250 scripts	Amudhan Stevo
Oct. 28-Nov. 1	Implement the desired parameters to work with flows and calculate energy/FIT	Compute servers	Stevo and Amudhan
Nov. 4-8	Continued work from previous week Begin thinking about how to scale FIT measurements	Compute servers Brains, class resources	Stevo and Amudhan Stevo and Amudhan
Nov. 11-15	Develop algorithms to scale FIT calculations for large circuits	Brains, class resources, papers	Stevo and Amudhan
Nov. 18-22	Implement algorithms into FIT calculation flow	Nothing new	Amudhan
Nov. 25-29	Thanksgiving		
Dec. 2-6	Compile results, write report	Nothing new	Stevo

[1] FIT is defined as failures in time, where 1 FIT equals 1 failure every billion hours of operation

References:

[1] D. Holcomb, W. Li and S. Seshia, *Design as you see FIT: System-level soft error analysis of sequential circuits*, DATE, 2009.