



Graduando da POLI-USP desenvolve nova arquitetura de roteamento robusta para FPGAs de memória não volátil.

São Paulo, 07 de dezembro de 2022.

Dispositivos conhecidos como Field-Programmable Gate Arrays (arranjo de portas lógicas programáveis em campo, em português) são excelentes alternativas para desenvolvimento e prototipagem de circuitos integrados e aceleradores para aplicações como criptografia e aprendizagem de máquina. Compostos por elementos de lógica, memória e roteamento, FPGAs podem ser programadas inúmeras vezes para reproduzir a funcionalidade de um circuito arbitrário que podem ir de um simples somador binário a um processador inteiro. A fim de melhorar a performance de dispositivos do gênero, Lucas Gaia, graduando em engenharia de computação pela Escola Politécnica da USP propôs uma nova arquitetura de roteamento tolerante a falhas.

Se inspirando em metodologias de projeto de sistemas tolerantes a falhas, o aluno Lucas Gaia elaborou uma nova arquitetura de roteamento onde um arranjo de duas células de memória, ao invés de apenas uma, controlam uma chave de roteamento das FPGAs, fornecendo robustez em caso de defeito. A justificativa para a robustez, segundo Lucas, é o uso de memórias programáveis baseadas em memristores ao invés de transistores, conforme o padrão da indústria para circuitos integrados. Apesar de já existirem processos industriais que produzam memórias baseadas em memristores, esses elementos são mais passíveis de defeito de produção do que transistores, que já possuem processos de fabricação bem amadurecidos.

A alternativa de memristores apresenta duas vantagens. A primeira é que a célula de memória baseada em memristores não precisa de energia para armazenar informação, diferente da memória atualmente utilizada pelas FPGAs, que exigem que elas sejam reprogramadas toda vez que o dispositivo é reiniciado; E a segunda é que memristores não dividem a mesma área que transistores no substrato, permitindo que mais área seja utilizada para funções lógicas e menos área seja utilizada para o roteamento ligando blocos lógicos, permitindo que circuitos mais complexos possam ser implementados em FPGAs desse tipo. Segundo a literatura o roteamento pode ser responsável por mais da metade da área utilizada e do tempo de latência em circuitos implementados em FPGAs.

Utilizando a linguagem de programação Python e uma coletânea de ferramentas para exploração de arquiteturas de FPGA chamada Verilog-to-Routing, Lucas implementou e simulou os novos elementos de roteamento com múltiplas taxas de defeito similares ao estado da arte na indústria. Seus resultados mostraram a nova arquitetura suporta até três vezes mais defeitos do que o estado da arte em memórias baseadas em memristores, apresentando chaveamento levemente mais rápido e ocupando aproximadamente 14,3% menos área, além de permitir que a programação do dispositivo persista mesmo com a FPGA desligada.

Integrante: Lucas Gaia de Castro

Professor Orientador: Bruno de Carvalho Albertini
