



UNIVERSIDADE D
COIMBRA



DEPARTAMENTO DE ENGENHARIA ELETROTECNICA E DE COMPUTADORES

Co-PROJETO HARDWARE E SOFTWARE

Projeto Final

Autor:

Gonçalo Bastos
Leonardo Cordeiro

Numero de Estudante:

2020238997
2020228071

January 19, 2024

1 Objetivo

Este trabalho visa ao projeto e implementação de operações aritméticas básicas em FPGA usando Computação Estocástica, em alternativa aos métodos convencionais de Computação Aritmética. Este projeto envolve a geração de streams de bits estocásticos e execução de operações lógicas para emular os princípios da computação estocástica. O foco será em demonstrar como operações aritméticas fundamentais podem ser realizadas eficientemente através de processos estocásticos, demonstrando as vantagens da computação estocástica na execução de operações aritméticas básicas em sistemas digitais modernos. Vamo-nos basear a utilizar como suporte as referencias enunciadas no fim.

2 Trabalho Desenvolvido

Conforme mencionado na seção anterior para alcançar o objetivo de implementar operações aritméticas básicas em FPGA utilizamos Computação Estocástica como alternativa à computação binária convencional na qual probabilidades digitalizadas são usadas para representar e processar informação. Sinais estocásticos são gerados por processos estocásticos contínuos no tempo, para este feito usamos um Left Feedback Shift Register (LFSR) para gerar sequências longas e pseudo aleatórias de '0's e '1's com um hardware relativamente simples e eficiente. O aumento linear na precisão dos cálculos estocásticos requer um aumento exponencial no comprimento do fluxo de bits, e a gama dinâmica da representação na CE também é limitada, desta forma nós definimos inicialmente um comprimento de 32 bits, com o objetivo futuro de aumentar este comprimento.

Realçamos o facto de o número pseudo aleatório tender para 0 caso a sua seed não seja realimentada, para tal existem soluções mais robustas com um "true random number generator", ou uma realimentação da semente no LFSR pseudo aleatório.

Para gerar o numero estocástico usamos um comparador, uma das entradas deste comparador será o bitstream pseudo aleatório gerado pelo LFSR, e a outra entrada será o numero binário, este proveniente do processador.

Como se pode ver na Fig 1 ,de forma a obter a representação binária do numero estocástico calculado foi usado um contador, a cada comparação é gerada o bit estocástico que vai ser passado para este mesmo contador de bits de '1', esse contador é definido com uma janela passada pelo processador, quando a janela é totalmente preenchida obtemos a representação binária do número estocástico inserido.

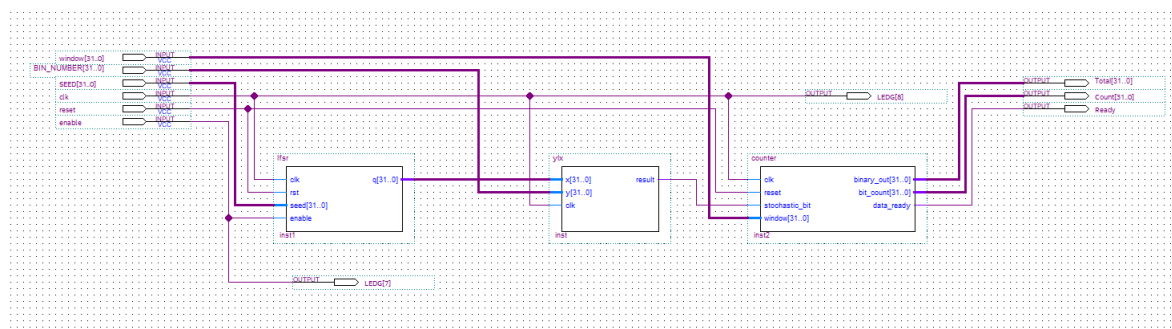


Figure 1: Binary to Stochastic and Stochastic to Binary RTL Design Blocks

A seguir temos o hardware completo do nosso projeto, que consiste em um clock, o bloco do processador NIOS II, e ainda o bloco B2S_S2B mencionado acima na figura 1.

A figura 3 mostra o output do projeto no terminal do *Eclipse*, onde mostra os bitstreams enviados para o processador como a seed para o LFSR e o numero binário que queremos passar para uma bitstream estocástica, e ainda definimos a janela de atuação para o contador(ex: 128 bits). No final

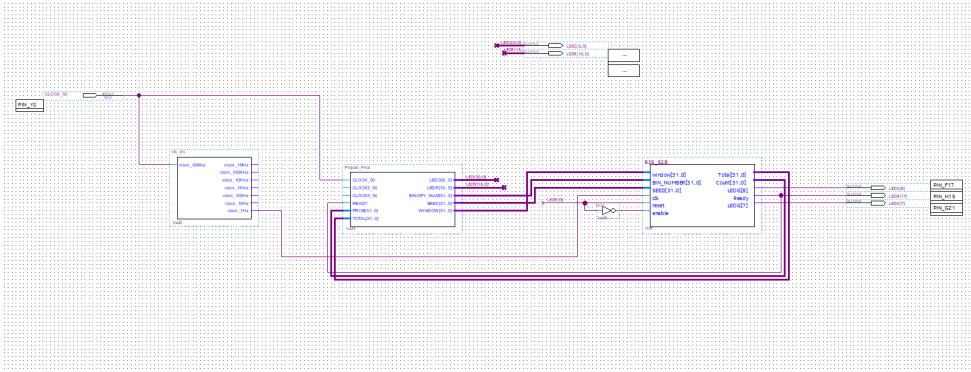


Figure 2: Main Block Project

da execução do programa é nos dada a probabilidade estocástica calculada, para o nosso exemplo o resultado obtido foi 14/128.

```

Problems Tasks Console Nios II Console Properties
New_configuration (1) - cable: USB-Blaster on localhost [USB-0] device ID: 1 instance ID: 0 name: jag_uart_0_jag

Seed Enviada:-> 4294679546
Numero Binario Enviado:-> 3568915764
Janela de calculo:-> 128 bits
Numero de Bits a 1: 14
Total Numero de Bits: 128
Probabilidade: 14 / 128
Programa Terminado!

```

Figure 3: Eclipse Output

2.1 Teste da implementação

Temos em anexo um vídeo da implementação a correr na placa FPGA. No qual inicialmente o LED vermelho '0' está ativado, ou seja é quando recebe a seed, depois de receber a seed é enviado o Número binário e a cada ciclo de relógio é feita uma comparação bitwise, ao fim da janela de atuação ser totalmente preenchida é ligado o LED vermelho '16' que mostra a conclusão do programa. O Output é mostrado no *Eclipse*, como demonstrado na *Figure 3*

3 Conclusão e Trabalho Futuro

Em conclusão, não atingimos as especificações definidas para este trabalho, nomeadamente a realização de operações aritméticas usando a computação estocástica. Porém, este trabalho foi um enorme desafio pelo qual consideramos ter alcançado um resultado satisfatório nomeadamente na perceção da computação estocástica e no âmbito da cadeira de CPHS as possibilidades de utilizar o Software lado a lado com o Hardware para a realização de operações de aritmética complexa para ser realizada apenas por software, esta conjugação pode trazer speedups de computação relativamente significativos bem como decréscimo no consumo energético. Futuramente, definimos concluir a realização de operações aritméticas usando a computação estocástica bem como estudar os desempenhos de consumo e de computação da utilização da FPGA juntamente com o Software para a realização destas operações aritméticas mais complexas para serem apenas resolvidas fazendo uso do Software.